HAL open science

Une analyse des exercices d'algorithmique et de programmation des DNB

Sylvie Alayrangues, Olivier Baudon, Sylvain Beauvoir, Emmanuel Beffara, Sébastien Daniel, Christophe Declercq, Aslı Grimaud, Anne Héam, Eddy Labidi, Philippe Marquet, et al.

To cite this version:

Sylvie Alayrangues, Olivier Baudon, Sylvain Beauvoir, Emmanuel Beffara, Sébastien Daniel, et al.. Une analyse des exercices d'algorithmique et de programmation des DNB. Commission inter-IREM informatique. 2023. hal-03996891v1

HAL Id: hal-03996891

https://hal.science/hal-03996891v1

Submitted on 20 Feb 2023 (v1), last revised 9 Feb 2024 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Une analyse des exercices d'algorithmique et de programmation des DNB

Commission inter-IREM informatique[1]

31 janvier 2023

[1]Liste des co-auteurs : Sylvie Alayrangues (IREM de Poitiers), Olivier Baudon (IREM de Bordeaux), Sylvain Beauvoir (IREM de Lyon), Emmanuel Beffara (IREM de Marseille / IREM de Grenoble), Sébastien Daniel (IREM de Lorraine), Christophe Declercq (IREM de Nantes / IREM de la Réunion), Asli Grimaud (IREM de Lilles) Anne Héam (IREM de Besançon), Philippe Marquet (IREM de Lille), Jean-Christophe Masseron (IREM de Paris), Antoine Meyer (IREM de Paris), Malika More (IREM de Clermont-Ferrand), Florence Nény (IREM de Marseille), Cécile Prouteau (IREM de Paris), Sylviane Schwer (IREM de Paris Nord), Jean-Marc Vincent (IREM de Grenoble), Emmanuel Volte (Université de Cergy-Pontoise)

Sommaire

1 Introduction **5**

1 Grille d'analyse des sujets de brevet 2017 . . . 6
2 Grille d'analyse des sujets de brevet 2018 . . . 9
3 Grille d'analyse des sujets de brevet 2019 . . . 12
4 Grille d'analyse des sujets de brevet 2020 . . . 15
5 Grille d'analyse des sujets de brevet 2021 . . . 18
6 Grille d'analyse des sujets de brevet 2022 . . . 21

2 Brevets 2017 **24**

1 Polynésie, juin 2017, série générale . . . 25
2 Polynésie, septembre 2017, série générale . . . 32
3 Pondichéry, mai 2017, série générale . . . 35
4 Sujet Zéro . . . 39
5 Amérique du Nord, juin 2017, série générale . . . 43
6 Asie, juin 2017, série générale . . . 48
7 Centres étrangers, juin 2017, série générale . . . 53
8 Métropole – La Réunion – Antilles – Guyane, juin 2017, série générale . . . 58
9 Métropole, La Réunion, Antilles-Guyanne, septembre 2017, série générale . . . 61
10 Amérique du Sud, novembre 2017, série générale . . . 66
11 Wallis et Futuna, décembre 2017, série générale . . . 73
12 Métropole – La Réunion , juin 2017, série professionnelle . . . 77
13 Polynésie, juin 2017, série professionnelle . . . 80
14 Métropole – La Réunion – Antilles – Guyane, septembre 2017, série professionnelle . . . 88
15 Polynésie, septembre 2017, série professionnelle . . . 92

3 Brevets 2018 **97**

1 Polynésie, juillet 2018, série générale . . . 98
2 Polynésie, septembre 2018, série générale . . . 102
3 Pondichéry, mai 2018, série générale . . . 107
4 Amérique du Nord, juin 2018, série générale . . . 120
5 Asie, juin 2018, série générale . . . 126
6 Centres Étrangers, juin 2018, série générale . . . 132
7 Métropole, juin 2018, série générale . . . 138
8 Métropole, septembre 2018, série générale . . . 142
9 Amérique du sud, novembre 2018, série générale . . . 147
10 Nouvelle Calédonie, décembre 2018, série générale . . . 152
11 Métropole – La Réunion – Antilles – Guyane, juin 2018, série professionnelle . . . 159
12 Polynésie, juin 2018, série professionnelle . . . 166
13 Polynésie, septembre 2018, série professionnelle . . . 172

4 Brevets 2019 **175**

1 Polynésie, juillet 2019, série générale . . . 176

2 Polynésie, septembre 2019, série générale . . . 182
3 Métropole La Réunion - Antilles-Guyane, septembre 2019, série générale . . . 185
4 Nouvelle Calédonie, mars 2019, série générale . . . 190
5 Grèce, juin 2019, série générale . . . 194
6 Centres étrangers, juin 2019, série générale . . . 198
7 Antilles-Guyane, juin 2019, série générale . . . 202
8 Métropole, juillet 2019, série générale . . . 206
9 Amérique du sud, novembre 2019, série générale . . . 211
10 Nouvelle-Calédonie, décembre 2019, série générale . . . 215
11 Amérique du Nord, juin 2019, série générale . . . 219
12 Asie, juin 2019, série générale . . . 222
13 Nouvelle Calédonie, mars 2019, série professionnelle . . . 226

5 Brevets 2020 230
1 Nouvelle-Calédonie, février 2020, série générale, exercice 2 . . . 231
2 Antilles-Guyane, septembre 2020, série générale . . . 234
3 Nouvelle-Calédonie, février 2020, série générale, exercice 8 . . . 238
4 Polynésie, septembre 2020, série générale . . . 242
5 Métropole, septembre 2020, série générale . . . 246
6 Nouvelle-Calédonie, décembre 2020, série générale . . . 249

6 Brevets 2021 253
1 Asie, juin 2021, série générale . . . 254
2 Centres étrangers, juin 2021, série générale . . . 257
3 Métropole, La Réunion, Antilles-Guyanne, juin 2021, série générale . . . 260
4 Polynésie, juin 2021, série générale . . . 263
5 Métropole, septembre 2021, série générale . . . 266
6 Amérique du Sud, novembre 2021, série générale . . . 269
7 Amérique du Nord, juin 2021, série générale . . . 274
8 Métropole, juin 2021, série professionnelle . . . 280

7 Brevets 2022 283
1 Polynésie, juin 2022, série générale, exercice 5 . . . 284
2 Asie, juin 2022, série générale . . . 288
3 Métropole, La Réunion, Antilles-Guyanne (Secours), juin 2022, série générale . . . 291
4 Métropole, La Réunion, Antilles-Guyanne, juin 2022, série générale . . . 294
5 Amérique du Nord, juin 2022, série générale . . . 297
6 Polynésie, juin 2022, série générale, exercice 4 . . . 300
7 Centres étrangers, juin 2022, série générale . . . 303
8 Polynésie, septembre 2022, série générale . . . 308
9 Métropole, La Réunion, Antilles-Guyanne, septembre 2022, série générale . . . 315
10 Amérique du Sud, novembre 2022, série générale . . . 318
11 Nouvelle-Calédonie, décembre 2022, série générale . . . 322
12 Métropole, juin 2022, série professionnelle . . . 325

Bibliographie 328

Index - Thèmes mathématiques **329**

Index - Thèmes informatique **330**

Index - Compétences informatiques **331**

Introduction

Cette brochure recense l'ensemble des exercices des épreuves du brevet ayant trait au thème « Algorithmique et programmation » du programme de mathématiques de cycle 4, de 2017 à 2022. Pour chaque exercice, sont identifiés les thèmes mathématiques et informatiques abordés ainsi que les compétences informatiques mises en jeu. En début de brochure figurent les grilles d'analyse par année, et en fin de brochure des index par thèmes mathématiques, thèmes informatiques et compétences informatiques. Tous les sujets contenant des programmes en Scratch ont été traduits au format Scratch 3 pour faciliter leur usage en classe. Les programmes correspondants sont disponibles sur le studio Scratch de la C3i. Vous êtes libres de les remixer pour les utiliser en classe.

Ce recueil est enrichi d'une analyse critique de nombreux exercices. Certains de ces exercices ne mettent en jeu des compétences spécifiquement informatiques que de manière très superficielle. Ils portent plutôt sur des compétences mathématiques « traditionnelles », liées par exemple à l'algèbre ou la géométrie. Notre analyse met en regard ces différentes compétences, et propose des adaptations de certains énoncés afin d'en permettre une exploitation plus riche ainsi que des exercices de prolongement.

Cette. publication fait suite à l'article publié dans Repère IREM Numéro 116 de juillet 2019 sur les sujets de 2017 [1]. Dans cette première publication, les exercices avaient été classés par thème :

— Programmes de calcul

— Tracé de figures : frises et rosaces

— Tracé de figures : triangles, spirales, et autres carrés

— Exercices sur les aspects événementiels

— Simulation en probabilités

Dans la présente brochure, les exercices sont classés par année. Nous espérions observer, au fil des ans, une évolution notable dans les thèmes informatiques mis en jeu ainsi que dans l'évaluation des compétences informatiques. Malheureusement, dans les exercices portant sur certains thèmes mathématiques (par exemple sur les programmes de calcul), nous observons au bout de 6 ans une forte redondance et un contenu informatique très pauvre. Certains exercices, proches d'exercices déjà rencontrés, ne nécessitent pas d'être de nouveau analysés et complétés. Par ailleurs, il serait intéressant de diversifier les approches et travailler d'autres compétences en proposant des exercices d'algorithmique ne reposant pas sur Scratch.

Quelques exercices sortent néanmoins du lot et abordent des thèmes plus originaux, comme la notion d'état. Nous nous sommes attardés sur ceux-ci. Certains sujets n'ont pas étés analysés car ils n'apportaient rien de nouveau. Ils ont étés insérés dans ce document pour y être indexés.

Les sources LaTeX des sujets de brevet utilisés dans ce document ont en grande partie été adaptées du site de l'APMEP [1]. Nous remercions ses contributeurs.

1. APMEP : Association des professeurs de mathématiques de l'enseignement public, `https://www.apmep.fr/Annales-du-Brevet-des-colleges`

1 Grille d’analyse des sujets de brevet 2017

no de sujet	nom de sujet
1	Polynésie 23 juin 2017
2	Polynésie 14 septembre 2017
3	Pondichéry 2 mai 2017
4	Sujet zéro du DNB 2017
5	Amérique du Nord 7 juin 2017
6	Asie 27 juin 2017
7	Centres étrangers 17 juin 2017
8	Métropole, Réu, Ant-Guy 29 juin 2017
9	Métropole, Réu, Ant-Guy 14 septembre 2017
10	Amérique du Sud 30 novembre 2017
11	Wallis et Futuna 2 décembre 2017
12	Métropole—La Réunion série professionnelle juin 2017
13	Polynésie série professionnelle juin 2017
14	Métropole—La Réunion série professionnelle septembre 2017
15	Polynésie série professionnelle septembre 2017

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Objets mobilisés																
spécifiques																
lutin																
état initial	10	0	0	0	1	0	1	1	1	1	1	0	1	1	1	1
Modification position, orientation etc.	12	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
multiplicité des lutins	2	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
scène (état du dessin)	7	0	0	0	1	0	1	1	1	1	1	1	0	0	0	0
général																
variables réservées (réponse, x…)	5	1	1	1	0	1	0	0	0	0	0	0	0	0	1	0
variables déclarées	11	0	2	4	0	0	0	0	1	1	1	0	0	1	0	1
capteur	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Instructions																
communication																
entrée	4	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1
sortie (affichage avec « dire »)	3	1	1	(3)+1	0	1	0	0	0	0	0	0	0	0	0	
activité du lutin																
déplacement	12	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
tracé	11	0	0	0	1	0	1	1	1	1	1	1	1	1	1	1
visibilité	3	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0
exécution interne																
variables																
affectation unique	2	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
changement de valeur	4	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1
chgt avec auto-référence	5	0	0	0	0	0	0	0	1	1	1	0	0	1	1	0
calcul	5	1	1	1	0	0	0	0	0	0	1	0	0	1	0	0

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Structures de contrôle séquentiel																
séquence	15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
conditionnelle																
réduite (si-alors)	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
complète (si-alors-sinon)	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
itération bornée	9	0	0	0	1	0	0	1	1	1	1	1	1	0	1	1
itération conditionnelle	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
imbrication de structures	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Procédures																
sans paramètres	9	0	0	0	1	1	1	1	1	1	1	1	0	0	1	0
avec paramètres	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gestion du temps																
attente	5	0	1	1	0	1	0	0	0	0	0	0	0	1	0	1
Structures de contrôle événementiel																
événements extérieurs	3	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
événements internes	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
messages		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Expressions																
expression arithmétique	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
expression logique (test de comparaison, d'égalité ...)	12	1	0	0	1	1	0	1	1	1	1	1	1	1	1	1
expression booléenne (ET, OU, NON …)	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Etat		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

2 Grille d’analyse des sujets de brevet 2018

no de sujet	nom de sujet
1	Polynésie 2 juillet 2018
2	Polynésie 10 septembre 2018
3	Pondichéry 3 mai 2018
4	Amérique du Nord 5 juin 2018
5	Asie 25 juin 2018
6	Centres étrangers 18 juin 2018
7	Métropole, Réunion 28 juin 2018
8	Métropole La Réunion Antilles-Guyane 20 septembre 2018
9	Amérique du Sud 30 novembre 2018
10	Nouvelle Calédonie 10 décembre 2018
11	Métropole La Réunion Antilles-Guyane série professionnelle juin 2018
12	Polynésie série professionnelle juin 2018
13	Polynésie série professionnelle septembre 2018

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12	13
Objets mobilisés														
spécifiques														
lutin														
état initial	8	0	0	0	1	1	1	1	1	1	1	1	0	0
Modification position, orientation etc.	9	0	0	1	1	1	1	1	1	1	1	1	0	0
multiplicité des lutins	0	0	0	0	0	0	0	0	0	0	0	0	0	0
scène (état du dessin)	4	0	0	0	1	1	1	1	0	0	0	0	0	0
général														
variables réservées (réponse, x…)	2	1	0	1	0	0	0	0	0	0	0	0	0	0
variables déclarées	8	1	1	1	1	0	0	1	1	0	1	0	1	0
capteur	1	0	0	0	0	0	0	0	0	0	0	0	0	1
Instructions														
communication														
entrée	1	1	0	0	0	0	0	0	0	0	0	0	0	0
sortie (affichage avec « dire »)	0	0	0	0	0	0	0	0	0	0	0	0	0	0
activité du lutin														
déplacement	9	0	0	1	1	1	1	1	1	1	1	1	0	0
tracé	7	0	0	0	1	1	1	1	1	1	1	0	0	0
visibilité	0	0	0	0	0	0	0	0	0	0	0	0	0	0
exécution interne														
variables														
affectation unique	3	1	0	1	0	0	0	0	0	0	0	0	1	0
changement de valeur	6	1	1	1	1	0	0	1	1	0	0	0	0	0
chgt avec auto-référence	7	1	1	1	1	0	0	0	1	0	1	0	1	0
calcul	7	1	1	1	1	0	0	1	1	0	1	0	0	0

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12	13
Structures de contrôle séquentiel														
séquence	7	0	0	1	1	1	1	1	0	0	0	1	1	0
conditionnelle														
réduite (si-alors)	1	0	0	1	0	0	0	0	0	0	0	0	0	0
complète (si-alors-sinon)	2	0	0	0	0	0	0	0	0	0	0	0	1	1
itération bornée	8	0	0	1	1	1	1	1	1	1	1	0	0	0
itération conditionnelle	1	0	1	0	0	0	0	0	0	0	0	0	0	0
imbrication de structures	1	0	0	0	0	0	0	0	0	0	0	0	0	1
Procédures														
sans paramètres	4	0	0	0	1	0	1	1	0	1	0	0	0	0
avec paramètres	2	0	1	0	0	0	0	0	1	0	0	0	0	0
Gestion du temps														
attente	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Structures de contrôle événementiel														
événements extérieurs	0	0	0	0	0	0	0	0	0	0	0	0	0	0
événements internes	0	0	0	0	0	0	0	0	0	0	0	0	0	0
messages		1	0	0	0	0	0	0	0	0	0	0	1	0
Expressions														
expression arithmétique	0	0	0	0	0	0	0	0	0	0	0	0	0	0
expression logique (test de comparaison, d'égalité ...)	2	0	0	1	0	0	0	0	0	0	0	0	1	0
expression booléenne (ET, OU, NON …)	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Etat		0	0	0	0	0	0	0	0	0	0	0	0	0

3 Grille d’analyse des sujets de brevet 2019

no de sujet	nom de sujet
1	Polynésie 1er juillet 2019
2	Polynésie 9 septembre 2019
3	Métropole La Réunion Antilles-Guyane 16 septembre 2019
4	Nouvelle Calédonie mars 2019
5	Grèce 18 juin 2019
6	Centres étrangers 14 juin 2019
7	Antilles-Guyane 27 juin 2019
8	Métropole La Réunion 1er juillet 2019
9	Amérique du Sud 26 novembre 2019
10	Nouvelle Calédonie 9 décembre 2019
11	Amérique du Nord 4 juin 2019
12	Asie 24 juin 2019
13	Nouvelle Calédonie série professionnelle mars 2019

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12	13
Objets mobilisés														
spécifiques														
lutin														
état initial	8	0	0	0	1	1	0	1	1	1	0	1	1	1
Modification position, orientation etc.	10	0	0	0	1	1	1	1	1	1	1	1	1	1
multiplicité des lutins	0	0	0	0	0	0	0	0	0	0	0	0	0	0
scène (état du dessin)	1	0	0	0	0	0	0	0	0	0	0	0	1	0
général														
variables réservées (réponse, x…)	6	1	1	0	0	1	1	0	0	0	0	1	0	1
variables déclarées	6	1	1	1	1	1	0	0	0	0	1	0	0	0
capteur	1	0	0	0	0	0	0	0	0	0	0	1	0	0
Instructions														
communication														
entrée	4	1	1	0	1	0	1	0	0	0	0	0	0	0
sortie (affichage avec « dire »)	3	1	1	0	0	0	0	0	0	0	0	1	0	0
activité du lutin														
déplacement	10	0	0	0	1	1	1	1	1	1	1	1	1	1
tracé	8	0	0	0	1	1	1	1	1	1	1	0	1	0
visibilité	0	0	0	0	0	0	0	0	0	0	0	0	0	0
exécution interne														
variables														
affectation unique	2	1	0	1	0	0	0	0	0	0	0	0	0	0
changement de valeur	4	1	0	1	1	0	1	0	0	0	0	0	0	0
chgt avec auto-référence	5	1	1	1	0	1	0	0	0	0	1	0	0	0
calcul	4	1	1	0	1	0	1	0	0	0	0	0	0	0

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12	13
Structures de contrôle séquentiel														
séquence	1	0	0	0	0	0	0	0	0	0	0	0	1	0
conditionnelle														
réduite (si-alors)	0	0	0	0	0	0	0	0	0	0	0	0	0	0
complète (si-alors-sinon)	3	0	0	1	0	0	0	0	1	0	0	1	0	0
itération bornée	8	0	0	1	0	1	1	1	1	1	1	0	1	0
itération conditionnelle	0	0	0	0	0	0	0	0	0	0	0	0	0	0
imbrication de structures	2	0	0	1	0	0	0	0	1	0	0	0	0	0
Procédures														
sans paramètres	5	0	0	1	0	1	0	1	1	1	0	0	0	0
avec paramètres	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gestion du temps														
attente	1	0	0	0	0	0	0	0	0	0	0	1	0	0
Structures de contrôle événementiel														
événements extérieurs	2	0	0	0	0	0	0	0	1	0	0	1	0	0
événements internes	1	0	0	0	0	0	0	0	0	0	0	1	0	0
messages		0	0	0	0	0	0	0	0	0	0	0	0	0
Expressions														
expression arithmétique	2	1	1	0	0	0	0	0	0	0	0	0	0	0
expression logique (test de comparaison, d'égalité ...)	2	0	0	1	0	0	0	0	1	0	0	0	0	0
expression booléenne (ET, OU, NON …)	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Etat		0	0	0	0	0	0	0	0	0	0	0	0	0

4 Grille d’analyse des sujets de brevet 2020

no de sujet	nom de sujet
1	Nouvelle-Calédonie 14 février 2020 exercice 2
2	Antilles-Guyane 14 septembre 2020
3	Nouvelle-Calédonie 14 février 2020 exercice 8
4	Polynésie 7 septembre 2020
5	Métropole-La Réunion 14 septembre 2020
6	Nouvelle-Calédonie 14 décembre 2020

Attendus	Total	1	2	3	4	5	6
Objets mobilisés							
spécifiques							
lutin							
état initial	4	0	0	1	1	1	1
Modification position, orientation etc.	4	0	0	1	1	1	1
multiplicité des lutins	0	0	0	0	0	0	0
scène (état du dessin)	0	0	0	0	0	0	0
général							
variables réservées (réponse, x…)	2	1	0	0	1	0	0
variables déclarées	3	1	1	0	0	0	1
capteur	0	0	0	0	0	0	0
Instructions							
communication							
entrée	0	0	0	0	0	0	0
sortie (affichage avec « dire »)	2	1	1	0	0	0	0
activité du lutin							
déplacement	4	0	0	1	1	1	1
tracé	4	0	0	1	1	1	1
visibilité	0	0	0	0	0	0	0
exécution interne							
variables							
affectation unique	2	1	1	0	0	0	0
changement de valeur	1	0	0	0	1	0	0
chgt avec auto-référence	3	0	1	0	1	0	1
calcul	3	1	1	0	1	0	0

Attendus	Total	1	2	3	4	5	6
Structures de contrôle séquentiel							
séquence	0	0	0	0	0	0	0
conditionnelle							
réduite (si-alors)	1	0	1	0	0	0	0
complète (si-alors-sinon)	0	0	0	0	0	0	0
itération bornée	5	0	1	1	1	1	1
itération conditionnelle	0	0	0	0	0	0	0
imbrication de structures	1	0	1	0	0	0	0
Procédures							
sans paramètres	4	0	1	1	1	1	0
avec paramètres	0	0	0	0	0	0	0
Gestion du temps							
attente	0	0	0	0	0	0	0
Structures de contrôle événementiel							
événements extérieurs	0	0	0	0	0	0	0
événements internes	0	0	0	0	0	0	0
messages		0	0	0	0	0	0
Expressions							
expression arithmétique	1	1	0	0	0	0	0
expression logique (test de comparaison, d'égalité ...)	1	0	1	0	0	0	0
expression booléenne (ET, OU, NON …)	0	0	0	0	0	0	0
Etat		0	0	0	0	0	0

5 Grille d’analyse des sujets de brevet 2021

no de sujet	nom de sujet
1	Asie 21 juin 2021
2	Centres étrangers 15 juin 2021
3	Métropole-La Réunion-Antilles -Guyanne 28 juin 2021
4	Polynésie 25 juin 2021
5	Métropole 13 septembre 2021
6	Amérique du Sud 23 novembre 2021
7	Amérique du nord 3 juin 2021
8	Métropole pro 28 juin 2021

Attendus	Total	1	2	3	4	5	6	7	8
Objets mobilisés									
spécifiques									
lutin									
état initial	2	0	0	0	1	0	1	0	0
Modification position, orientation etc.	3	0	0	0	1	1	1	0	0
multiplicité des lutins	0	0	0	0	0	0	0	0	0
scène (état du dessin)	0	0	0	0	0	0	0	0	0
général									
variables réservées (réponse, x…)	3	1	1	1	0	0	0	0	0
variables déclarées	4	0	1	1	1	0	0	0	1
capteur	0	0	0	0	0	0	0	0	0
Instructions									
communication									
entrée	3	1	1	1	0	0	0	0	0
sortie (affichage avec « dire »)	3	1	1	1	0	0	0	0	0
activité du lutin									
déplacement	3	0	0	0	1	1	1	0	0
tracé	3	0	0	0	1	1	1	0	0
visibilité	0	0	0	0	0	0	0	0	0
exécution interne									
variables									
affectation unique	3	0	1	1	0	0	0	0	1
changement de valeur	1	0	0	0	1	0	0	0	0
chgt avec auto-référence	2	0	0	0	1	0	0	0	1
calcul	3	1	1	0	1	0	0	0	0

Attendus	Total	1	2	3	4	5	6	7	8
Structures de contrôle séquentiel									
séquence	0	0	0	0	0	0	0	0	0
conditionnelle									
réduite (si-alors)	0	0	0	0	0	0	0	0	0
complète (si-alors-sinon)	1	1	0	0	0	0	0	0	0
itération bornée	4	0	0	0	1	1	1	0	1
itération conditionnelle	0	0	0	0	0	0	0	0	0
imbrication de structures	0	0	0	0	0	0	0	0	0
Procédures									
sans paramètres	2	0	0	0	1	0	1	0	0
avec paramètres	0	0	0	0	0	0	0	0	0
Gestion du temps									
attente	1	0	0	0	0	0	0	0	1
Structures de contrôle événementiel									
événements extérieurs	0	0	0	0	0	0	0	0	0
événements internes	0	0	0	0	0	0	0	0	0
messages		0	0	0	0	0	0	0	0
Expressions									
expression arithmétique	3	1	1	0	1	0	0	0	0
expression logique (test de comparaison, d'égalité ...)	1	1	0	0	0	0	0	0	0
expression booléenne (ET, OU, NON …)	0	0	0	0	0	0	0	0	0
Etat		0	0	0	0	0	0	1	0

6 Grille d’analyse des sujets de brevet 2022

no de sujet	nom de sujet
1	Polynésie 23 juin 2022 ex 5
2	Asie 20 juin 2022
3	Métropole-La Réunion-Antilles -Guyanne 30 juin 2022 (Secours)
4	Métropole-La Réunion-Antilles -Guyanne 30 juin 2022
5	Amérique du nord 3 juin 2022
6	Polynésie 23 juin 2022 ex 4
7	Centres étrangers 14 juin 2022
8	Polynésie 6 septembre 2022
9	Métropole — Antilles-Guyane 12 septembre 2022
10	Amérique du Sud 16 novembre 2022
11	Nouvelle-Calédonie 13 décembre 2022
12	Métropole Antilles-Guyane série professionnelle 30 juin 2022

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12
Objets mobilisés													
spécifiques													
lutin													
état initial	6	1	0	0	0	1	0	1	1	0	1	1	0
Modification position, orientation etc.	7	1	0	0	0	1	0	1	1	1	1	1	0
multiplicité des lutins	0	0	0	0	0	0	0	0	0	0	0	0	0
scène (état du dessin)	0	0	0	0	0	0	0	0	0	0	0	0	0
général													
variables réservées (réponse, x…)	6	0	1	1	1	0	0	0	0	0	1	1	1
variables déclarées	5	0	1	1	1	0	0	0	1	0	0	0	1
capteur	0	0	0	0	0	0	0	0	0	0	0	0	0
Instructions													
communication													
entrée	4	0	1	1	1	0	0	0	0	0	0	0	1
sortie (affichage avec « dire »)	4	0	1	1	1	0	0	0	0	0	0	0	1
activité du lutin													
déplacement	7	1	0	0	0	1	0	1	1	1	1	1	0
tracé	7	1	0	0	0	1	0	1	1	1	1	1	0
visibilité	0	0	0	0	0	0	0	0	0	0	0	0	0
exécution interne													
variables													
affectation unique	4	0	1	1	1	0	0	0	0	0	0	0	1
changement de valeur	1	0	0	0	0	0	0	0	1	0	0	0	0
chgt avec auto-référence	3	0	0	1	1	0	0	0	1	0	0	0	0
calcul	5	0	1	1	1	0	1	0	0	0	0	0	1

Attendus	Total	1	2	3	4	5	6	7	8	9	10	11	12
Structures de contrôle séquentiel													
séquence	0	0	0	0	0	0	0	0	0	0	0	0	0
conditionnelle													
réduite (si-alors)	1	0	1	0	0	0	0	0	0	0	0	0	0
complète (si-alors-sinon)	2	0	1	0	0	1	0	0	0	0	0	0	0
itération bornée	6	1	0	0	0	1	0	1	1	0	1	1	0
itération conditionnelle	0	0	0	0	0	0	0	0	0	0	0	0	0
imbrication de structures	2	0	0	0	0	1	0	0	0	0	0	1	0
Procédures													
sans paramètres	4	0	0	0	0	1	0	1	0	0	1	1	0
avec paramètres	0	0	0	0	0	0	0	0	0	0	0	0	0
Gestion du temps													
attente	1	0	0	0	0	0	0	0	0	0	0	1	0
Structures de contrôle événementiel													
événements extérieurs	0	0	0	0	0	0	0	0	0	0	0	0	0
événements internes	0	0	0	0	0	0	0	0	0	0	0	0	0
messages		0	0	0	0	0	0	0	0	0	0	0	0
Expressions													
expression arithmétique	4	0	1	1	1	0	0	0	0	0	0	0	1
expression logique (test de comparaison, d'égalité ...)	2	0	1	0	0	1	0	0	0	0	0	0	0
expression booléenne (ET, OU, NON …)	0	0	0	0	0	0	0	0	0	0	0	0	0
Etat	0	0	0	0	0	0	0	0	0	0	0	0	0

Brevets 2017

1 Polynésie, juin 2017, série générale

Références

Série : générale

Académie : Polynésie

Date : juin 2017

Exercice 5 - 7 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable réservée
- communication
- calcul
- branchement conditionnel
- évènement sans message

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

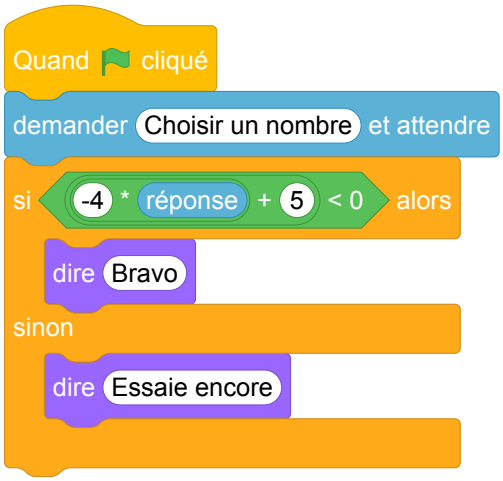
Énoncé - Polynésie, juin 2017

On considère le programme de calcul suivant :

- Choisir un nombre.
- Le multiplier par -4.
- Ajouter 5 au résultat.

1. Vérifier que lorsque l'on choisit -2 avec ce programme, on obtient 13.
2. Quel nombre faut-il choisir au départ pour obtenir -3 ?
3. Salomé fait exécuter le script suivant :

Script



   (a) Quelle sera la réponse du lutin si elle choisit le nombre 12 ?
   (b) Quelle sera la réponse du lutin si elle choisit le nombre -5 ?

4. Le programme de calcul ci-dessus peut se traduire par l'expression littérale $-4x+5$ avec x représentant le nombre choisi.
   Résoudre l'inéquation suivante : $-4x+5<0$.
5. À quelle condition, portant sur le nombre choisi, est-on certain que la réponse du lutin sera « Bravo » ?

Correction

Fichier scratch : `sg_polynesie_2017_juin.sb3`
Lien vers le fichier Scratch en ligne : `scratch.mit.edu/projects/362422190/`

1. On a $(-2) \times (-4) = 8$ et $8 + 5 = 13$.
2. On peut revenir au nombre de départ :
   $-3 - 5 = -8$ puis $\dfrac{-8}{-4} = 2$.
3. (a) On a $-4 \times 12 = -48$ et $-48 + 5 = -43 < 0$. Le lutin dira « Bravo ».
   (b) On a $-4 \times -5 = 20$ et $20 + 5 = 25 > 0$. Le lutin dira « Essaie encore ».
4. $-4x + 5$ avec x représentant le nombre choisi.
   Si $-4x + 5 < 0$, alors $5 < 4x$ puis $\dfrac{5}{4} < x$ ou $x > \dfrac{5}{4}$.
   Les nombres solutions sont les supérieurs à 1,25.
5. Le lutin dira « Bravo »dès que l'on choisira un nombre supérieur à 1,25.

Analyse

Cet exercice a le mérite de faire travailler le lien entre un programme de calcul, effectuant une succession d'opérations une par une, et une expression littérale équivalente. Cependant, le travail de transcription proprement dit n'est pas à la charge de l'élève ici. De plus, il est demandé à l'élève d'admettre l'équivalence entre le programme de calcul et l'expression $-4x + 5$ à la question 4, alors que cette information aurait pu servir à justifier la création du programme Scratch de la question 3.

On remarque une nette séparation entre les questions portant sur les aspects algébriques (questions 1, 2 et 4) et les questions portant sur la programmation (3 et 5). L'originalité du programme Scratch proposé est de ne pas constituer une transcription exacte du programme de calcul, du fait de la présence d'une expression composée et d'un bloc conditionnel. Malgré tout, comme dans la majorité des exercices du DNB 2017, le niveau de connaissances et compétences évaluées en algorithmique et programmation reste très modeste (compréhension élémentaire du fonctionnement général de Scratch, du bloc conditionnel, du bloc « demander », etc.).

Plus problématique, et typique de certains exercices portant sur des programmes, la formulation de la question 5 n'interdit pas a priori une réponse triviale du type « La réponse sera « Bravo » si $-4 *$ réponse $+ 5 < 0$ », ce qui n'est évidemment pas la réponse attendue. Une reformulation possible serait : « En déduire les nombres pour lesquels la réponse du lutin sera « Bravo ». », mais cela reste peu ambitieux en terme de niveau de connaissances mis en œuvre.

Propositions

Traduire un programme de calcul en Scratch

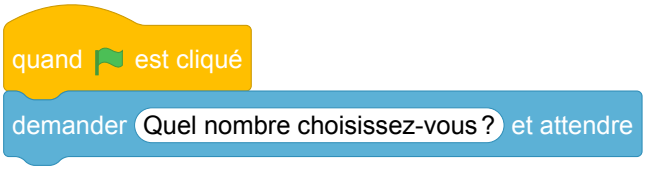
On fournit des blocs individuels, numérotés, chacun réalisant une affectation à une variable résultat, et un programme de calcul en français. Le but de l'exercice est de composer un programme Scratch équivalent au programme de calcul.

Exemple d'énoncé de type brevet.

Thomas utilise le programme de calcul ci-dessous :

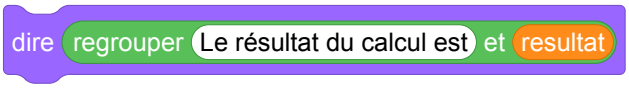
- Choisir un nombre.
- Le Multiplier par 3.
- Ajouter 2 au résultat.

1. Thomas choisit 2, quel résultat obtient-il ?
2. Leïla décide de traduire ce programme de calcul en langage Scratch, pour vérifier le calcul de Thomas. Voici le début de son programme :

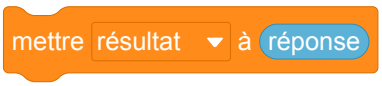


On rappelle qu'après l'exécution de ces deux blocs, la variable réponse contient une valeur saisie au clavier. Ordonnez et complétez les blocs suivants pour aider Leïla à terminer son programme :

1

2 

3

4 

Notions travaillées.

— Programmation : notion de variable informatique, d'expression, d'affectation, séquences d'instructions.

— Mathématiques : utiliser un programme de calcul.

Variantes. En salle machine, il est possible de proposer le même exercice sans fournir les blocs de base. Une autre variante, également adaptée à un contrôle sur table, consiste à proposer plusieurs programmes en Scratch et à demander à l'élève de choisir celui qui correspond au programme de calcul.

Décomposer une expression arithmétique en un programme de calcul

Ce type d'exercice consiste à fournir à l'élève une expression littérale contenant des nombres entiers, plusieurs opérations arithmétiques, éventuellement des parenthèses et une indéterminée *x*. La consigne est de concevoir un programme Scratch calculant la valeur de l'expression pour toute valeur choisie de *x*. L'énoncé peut éventuellement fournir une liste de blocs individuels à ordonner, ou une liste de programmes dont il faut déterminer lequel résout le problème.

Il est à noter que même si la décomposition du programme de calcul en calculs « élémentaires »n'est pas suggérée ou imposée par l'énoncé, l'interface Scratch impose néanmoins à l'élève-programmeur de choisir et de combiner les opérations arithmétiques de façon appropriée, ce qui correspond bien au savoir-faire visé par cet exercice.

Réciproquement, on peut fournir un programme Scratch réalisant un calcul complexe et demander à l'élève de rédiger le programme de calcul correspondant, ou de décomposer le programme en opérations élémentaires, ou de dessiner la structure de l'expression sous la forme d'un arbre dont chaque nœud interne correspond à un opérateur et dont chaque feuille correspond à un littéral.

Notions travaillées

— Programmation : notion de variable informatique, d'expression, d'évaluation d'une expression, d'initialisation.

— Mathématiques : identifier la nature de l'expression (addition, multiplication, etc.), ce qui revient à identifier la racine de l'arbre syntaxique de l'expression ; connaître la priorité des opérations.

Résoudre une équation par différentes méthodes

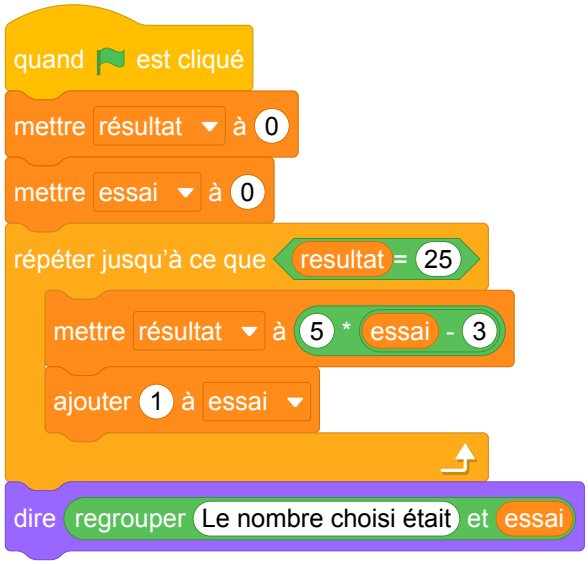
Ce type d'exercice vise à faire travailler la résolution d'équations du premier degré par différentes méthodes : essai / erreur, approche de type dichotomie, recherche exhaustive (sous l'hypothèse d'existence de solution entière), passage par la fonction réciproque, méthode algébrique. On fournit une expression littérale contenant des nombres entiers, plusieurs opérations arithmétiques, éventuellement des parenthèses et une variable *x*. La consigne est de concevoir un programme Scratch permettant de déterminer la valeur de *x* pour laquelle l'expression prend une valeur donnée.

Exemple d'énoncé

Ryan utilise le programme de calcul suivant :

- Choisir un nombre.
- Soustraire 3 au nombre choisi.
- Multiplier par 5 le résultat précédent.

1. Ryan obtient 25, quel était le nombre départ ?
2. Mathilde a écrit le programme Scratch ci-dessous pour retrouver le résultat de Ryan.



   Mathilde obtient 9, et dit à Ryan qu'il s'est trompé. Qui a raison ? Justifier.

3. Jérémie cherche de son côté, il propose le bloc suivant pour résoudre le problème directement :

   Compléter ce bloc.

4. On donne un programme de calcul avec des paramètres *a* et *b* du type :

   - Choisis un nombre ;
   - Multiplie par *a* le nombre choisi ;
   - Ajoute *b* au résultat précédent.

   Écrire un programme qui, lorsque l'on donne le résultat du calcul, retrouve le nombre choisi au départ.

Variantes et prolongements. On peut décliner cet exercice en choisissant d'autres variables didactiques (nombres décimaux, entiers négatifs, nombres rationnels) afin de différencier, et de travailler l'ensemble des techniques de résolution possibles.

Par exemple, l'approche proposée dans la question 2 ne fonctionne que si la solution recherchée est un entier strictement positif. Il faut donc envisager les cas où :

— la solution recherchée est un entier négatif – dans ce cas il faudra concevoir un autre type de boucle ;

— la solution recherchée n'est pas entière – dans ce cas, il faudra modifier la condition d'arrêt, et rechercher un encadrement de la solution, avant éventuellement de rechercher un encadrement plus précis.

2 Polynésie, septembre 2017, série générale

Références

Série : générale

Académie : Polynésie

Date : septembre 2017

Exercice 6 - 6 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable déclarée
- communication
- activité du lutin
- calcul
- évènement sans message
- capteurs

Compétences travaillées

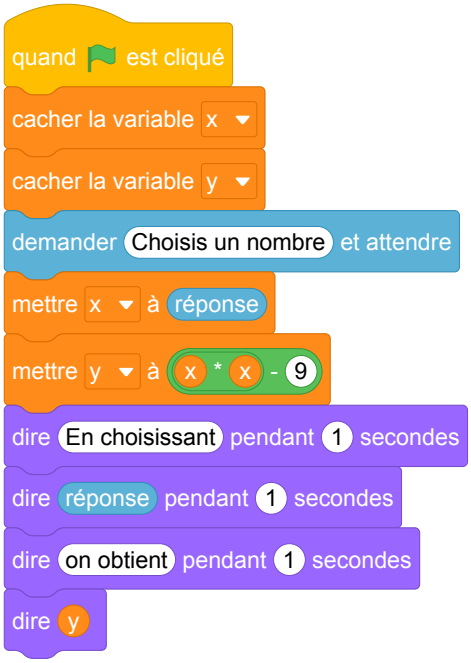
- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	2
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, septembre 2017

La figure ci-après est la copie d'écran d'un programme réalisé avec le logiciel « Scratch ».

1. Montrer que si on choisit 2 comme nombre de départ, alors le programme renvoie -5.
2. Que renvoie le programme si on choisit au départ :
   (a) le nombre 5 ?
   (b) le nombre -4 ?
3. Déterminer les nombres qu'il faut choisir au départ pour que le programme renvoie 0.



Correction

Fichier scratch : `sg_polynesie_2017_septembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/521442900`

1. Avec $x = 2$, $y = x^2 - 9 = 4 - 9 = -5$.
2. (a) si $x = 5$, $y = 5^2 - 9 = 25 - 9 = 16$;

   (b) si $x = -4$, $y = (-4)^2 - 9 = 16 - 9 = 7$.
3. Il faut que $y = x^2 - 9 = 0$, soit $(x+3)(x-3) = 0$

   ou $\begin{cases} x+3 = 0 \\ \text{ou} \\ x-3 = 0 \end{cases}$ et finalement $\begin{cases} x = -3 \\ \text{ou} \\ x = 3 \end{cases}$

   Pour obtenir 0 à la fin du programme on peut choisir au départ -3 ou 3.

Analyse

Cet exercice peut être associé à un programme de calcul ou à la production de l'expression de la fonction qui à x associe $x^2 - 9$. Le choix du nom de variable (y) nous fait tendre vers le domaine des fonctions. Le programme permet alors de calculer l'image de la valeur saisie par l'utilisateur (récupérée dans la variable réponse puis mémorisée dans (x)) et de la mémoriser dans la variable (y).

Du point de vue mathématique, les deux premières questions demandent de calculer trois images par la fonction f. Du point de vue informatique, c'est la connaissance de la manière dont on programme une expression arithmétique combinée qui permet d'identifier l'expression produite (l'opérateur binaire de multiplication imbriqué dans celui de la soustraction).

Quant à la dernière question, elle demande de résoudre l'équation $x^2 - 9 = 0$ et d'ainsi identifier les antécédents de 0 par la fonction f. L'élève peut soit procéder par essai-erreur, soit résoudre l'équation équivalente $x^2 = 9$ ou encore factoriser l'expression $x^2 - 9$ en $(x-3)(x+3)$. Le saut de difficulté est intrinsèque aux mathématiques.

Concernant la formulation des questions de cet exercice, les auteurs utilisent l'expression *ce que renvoie le programme* pour faire référence au contenu de la variable (y), ou à ce que l'exécution de ce programme affiche. Cela n'est pas judicieux.

Du point de vue informatique, la seule compétence évaluée est l'exécution à la main d'un algorithme, et dans une très faible mesure seulement puisque l'algorithme proposé ne contient ni condition ni boucle. Pour finir, les instructions [cacher la variable x] et [cacher la variable y] relèvent de la connaissance des spécificités du langage Scratch et n'apportent rien à l'exercice puisque la scène n'est pas visible.

3 Pondichéry, mai 2017, série générale

Références

Série : générale

Académie : Pondichéry

Date : mai 2017

Exercice 3 - 7 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable réservée
- variable déclarée
- calcul
- capteurs

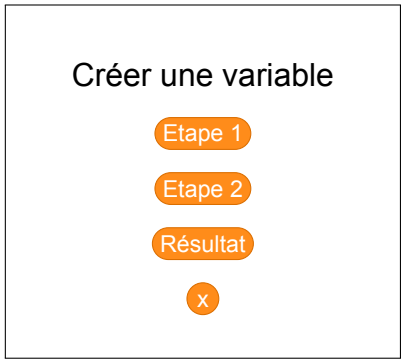
Compétences travaillées

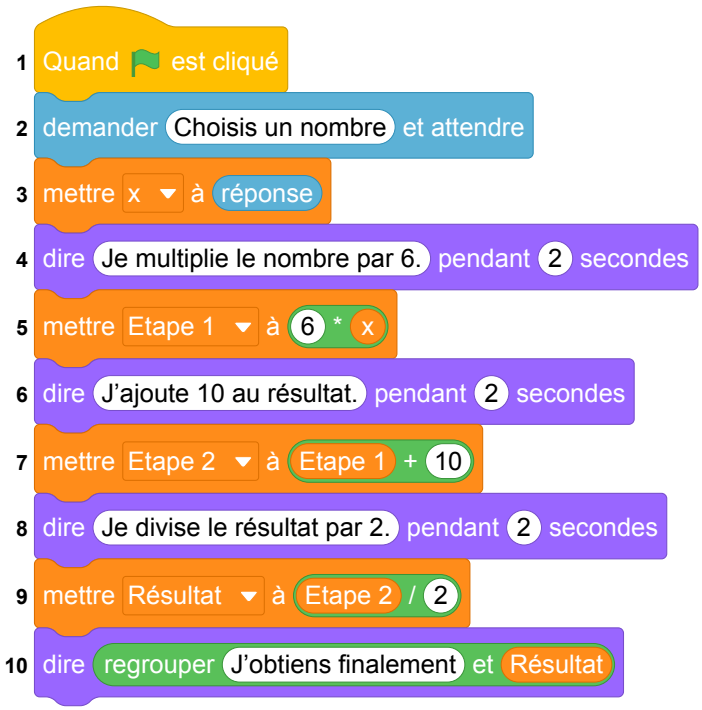
- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	4
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	(3)+1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Pondichéry, mai 2017

On considère le programme de calcul ci-contre dans lequel x, Etape 1, Etape 2 et Résultat sont quatre variables.





1. (a) Julie a fait fonctionner ce programme en choisissant le nombre 5. Vérifier que ce qui est dit à la fin est : « J'obtiens finalement 20 ».

   (b) Que dit le programme si Julie le fait fonctionner en choisissant au départ le nombre 7 ?

2. Julie fait fonctionner le programme, et ce qui est dit à la fin est : « J'obtiens finalement 8 ». Quel nombre Julie a-t-elle choisi au départ ?

3. Si l'on appelle x le nombre choisi au départ, écrire en fonction de x l'expression obtenue à la fin du programme, puis réduire cette expression autant que possible.

4. Maxime utilise le programme de calcul ci-dessous :

   - Choisir un nombre.
   - Lui ajouter 2.
   - Multiplier le résultat par 5.

   Peut-on choisir un nombre pour lequel le résultat obtenu par Maxime est le même que celui obtenu par Julie ?

Correction

Fichier scratch : `sg_Pondichery_2017_mai.sb3`

1. (a) $x = 5$
   - Ligne 5 : Etape 1 prend la valeur de $5 \times 6 = 30$
   - Ligne 7 : Etape 2 prend la valeur de $30 + 10 = 40$
   - Ligne 9 : Résultat prend la valeur de $40 \div 2 = 20$
   - dire « J'obtiens finalement 20 ».

   (b) $x = 7$
   - Ligne 5 : Etape 1 prend la valeur de $7 \times 6 = 42$
   - Ligne 7 : Etape 2 prend la valeur de $42 + 10 = 52$
   - Ligne 9 : Résultat prend la valeur de $52 \div 2 = 26$
   - dire « J'obtiens finalement 26 ».
2. Pour retrouver le nombre du départ il faut « remonter »l'algorithme, d'où
   - Ligne 9 : Résultat qui vaut 8 entraîne que Etape 2 vaut $8 \times 2 = 16$
   - Ligne 7 : Etape 2 vaut 16 ce qui nous donne que Etape 1 vaut $16 - 10 = 6$
   - Ligne 5 : Etape 1 vaut 6 ce qui nous donne que x vaut $6 \div 6 = 1$
   - Julie a choisi le nombre 1.
3. - Ligne 5 : Etape 1 prend la valeur de $6 \times x = 6x$
   - Ligne 7 : Etape 2 prend la valeur de $6x + 10$
   - Ligne 9 : Résultat prend la valeur de $(6x + 10) \div 2$

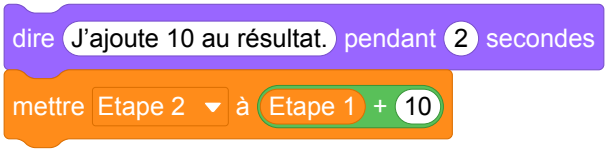
   Réduction : $(6x + 10) \div 2 = \dfrac{6x + 10}{2} = \dfrac{2(3x + 5)}{2} = 3x + 5$, ou encore
   $= (6x + 10) \div 2 = 6x \div 2 + 10 \div 2 = 3x + 5$.
4. Soit x le nombre choisi.
   Le programme de Maxime donne : $(x + 2) \times 5 = 5(x + 2) = 5x + 10$.
   On veut que $5x + 10 = 3x + 5$, d'où
   $$\begin{aligned} 5x + 10 &= 3x + 5 \\ 5x + 10 - 10 &= 3x + 5 - 10 \\ 5x &= 3x - 5 \\ 5x - 3x &= 3x - 5 - 3x \\ 2x &= -5 \\ 2x \div 2 &= -5 \div 2 \\ x &= -2,5 \end{aligned}$$
   Donc nous pouvons choisir le nombre $-2,5$ pour que le résultat obtenu par Maxime soit le même que celui obtenu par Julie.

Analyse

L'intention affichée de cet exercice est de questionner le fonctionnement de la transcription en Scratch d'un programme de calcul. Nous pouvons distinguer des instructions de deux natures différentes : des instructions d'affichage en bleu et des instructions d'affectation de variables en orange. Il semble qu'au-delà de cette concomitance, cet exercice, dans sa formulation, permet d'éviter de tester la compréhension de l'attendu affiché par le sujet « questionner le déroulement des instructions d'affectation ». En effet la simple lecture des instructions d'affichage en bleu permet de répondre à l'ensemble des questions sans s'interroger sur le contenu des variables présentes dans l'énoncé et le programme.

D'autre part, les différentes variables qui apparaissent dans l'exercice et leurs noms illustrent les difficultés liées à l'enseignement simultané des notions de variables mathématiques et informatiques au collège. Le programme utilise quatre variables informatiques, dont trois portent des noms riches, et une s'appelle seulement x. On note que ce *x* est le nom de la variable mathématique utilisée dans l'exercice. Les variables Étape 1 et Étape 2 contiennent des résultats intermédiaires, ce que leur nom, bien que riche, ne rend pas explicite. Peut-être aurait-il été préférable de choisir par exemple Résultat 1, Résultat 2 et Résultat Final par exemple, ou même de réutiliser plusieurs fois une même variable Résultat, ce qui permettrait en plus de vérifier la compréhension de la notion de variable informatique.

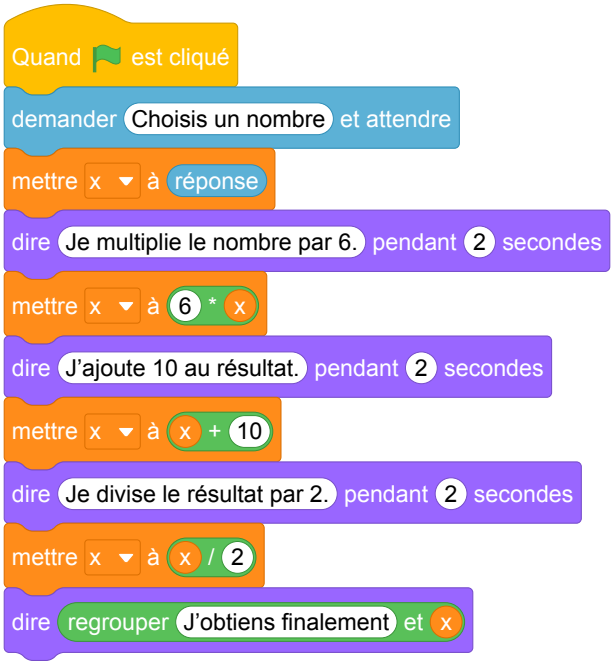
En outre, les instructions « dire » intermédiaires décrivent le calcul à la façon d'un programme de calcul ordinaire. En particulier, elles utilisent le mot « résultat » pour faire référence au résultat intermédiaire précédent :



Ceci peut de nouveau entraîner de la confusion vis-à-vis des variables informatiques utilisées, qui portent des noms différents (Étape 1, etc.).

Propositions

Au lieu du programme utilisé dans l'énoncé, nous pouvons proposer le même exercice avec le programme suivant qui utilise qu'une seule variable créée x.



Ce programme utilise la variable x au sens informatique, en effet, à chaque étape de calcul, la valeur de la variable est modifiée.

4 Sujet Zéro

Références

Série : générale

Académie : *

Date : 2017

Exercice 4

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- frise
- déplacement
- translation

Thèmes informatiques

- activité du lutin
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	1
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	1
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

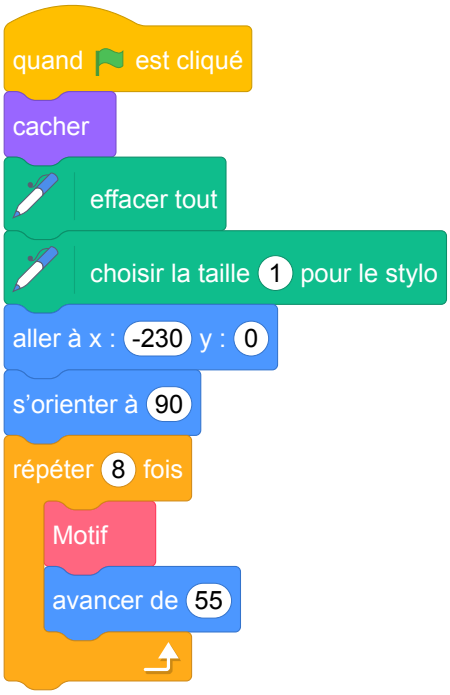
Énoncé - Sujet zéro, 2017

1. Pour réaliser la figure ci-dessus, on a défini un motif en forme de losange et on a utilisé l'un des deux programmes A et B ci-dessous.

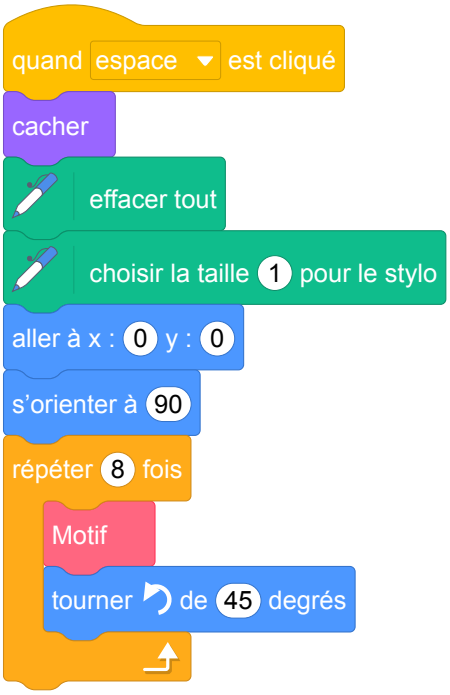
   Déterminer lequel et indiquer par une figure à main levée le résultat que l'on obtiendrait avec l'autre programme.

Motif

Programme A



Programme B



2. Combien mesure l'espace entre deux motifs successifs ?
3. On souhaite réaliser la figure ci-dessous :

Pour ce faire, on envisage d'insérer l'instruction

dans le programme utilisé à la question 1. Où faut-il insérer cette instruction ?

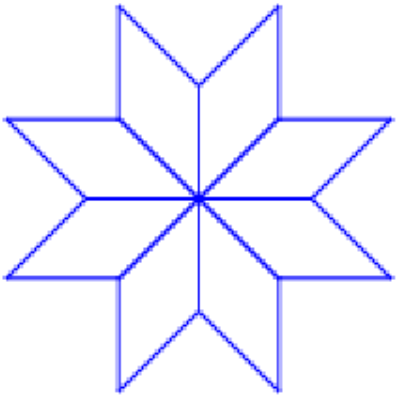
Correction

Fichier Scratch : `sujet_zero_2017.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/635978802/`

1. Le dessin proposé est obtenu avec le programme A. En effet il s'agit d'une frise et c'est le bloc *avancer de 55* qui se trouve dans la boucle répéter qui permet de translater le motif.

   Le programme B permet d'obtenir le dessin suivant puisque l'exécution du bloc *motif* ramène le lutin au point de départ et dans son orientation initiale et qu'à la fin de chaque itération, il tourne sur lui-même de 45 degrés :



2. L'espace est de 55-40=15 unités Scratch (longueur dont on avance à chaque étape - longueur d'un côté puisque l'exécution du bloc *motif* ramène le lutin au point de départ et dans son orientation initiale).

3. Il faut insérer l'instruction

à l'intérieur de la boucle répéter 8 fois, après le bloc motif, avant ou après le bloc *avancer de 55*.

Analyse

Ici, répondre à la première question relève de connaissances sur l'environnement Scratch, ici, la scène dont les points ont des abscisses comprises entre -240 et 240. Par ailleurs même si cette connaissance est absente, la présence du tourner de 45° dans la boucle permet elle aussi d'éliminer le programme B. L'absence complète de déplacement en dehors du tracé du motif est une indication forte. La discrimination des deux programmes ne peut alors attester de connaissances informatiques.

Par ailleurs, la formulation même de la question est problématique. En effet il y a une confusion entre le motif (le losange) produit et le nouveau bloc que l'on définit le bloc motif.

En ce qui concerne la question 3, une réponse de formulation vague telle que « dans la boucle » est acceptable puisqu'en effet, cette instruction peut être placée à n'importe quel endroit de la boucle (trois positions possibles). L'instruction peut aussi être placée au début ou à la fin du jeu d'instructions constituant la définition du bloc motif (avant ou après les blocs définissant l'état du stylo). Il y a alors au moins sept réponses correctes différentes à cette question. Cependant, positionner l'instruction dans la définition du bloc motif n'atteste pas nécessairement de la compréhension de la notion de boucle. En effet une formulation du type « chaque motif doit faire augmenter la taille du stylo » est suffisante.

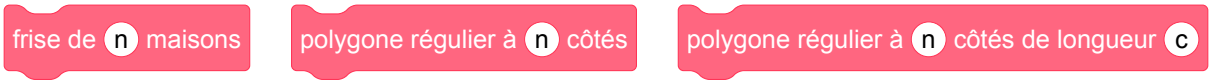
Propositions

Blocs à paramètres

Même si les blocs personnalisés de Scratch ne sont pas, à proprement parler, des *fonctions* au sens usuel de ce mot en programmation, puisqu'ils ne fournissent pas un résultat mais produisent uniquement des effets de bord (comme nous l'avons décrit ci-dessus), ils permettent de commencer à initier les élèves à la notion de fonction, en introduisant notamment le concept de paramètre. Les activités tournant autour du dessin et de la géométrie sont un cadre idéal pour cela.

Par exemple, une activité pourrait proposer aux élèves de définir des blocs dessinant des carrés de côtés de différentes longueurs, et leur faire observer qu'il est fastidieux et source d'erreurs de reproduire le même morceau de programme plusieurs fois. L'activité proposerait ensuite d'introduire la notion de bloc à paramètre, et de simplement définir un bloc carré de côté c.

Voici quelques autres exemples de blocs à paramètres qu'il est envisageable de définir avec les élèves :



5 Amérique du Nord, juin 2017, série générale

Références

Série : générale

Académie : Amérique du Nord

Date : juin 2017

Exercice 5 - 4,5 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- repérage
- déplacement
- calcul sans programme
- proportionnalité

Thèmes informatiques

- scène
- variable reservée
- activité du lutin
- bloc sans parametre
- évènement sans message
- capteurs

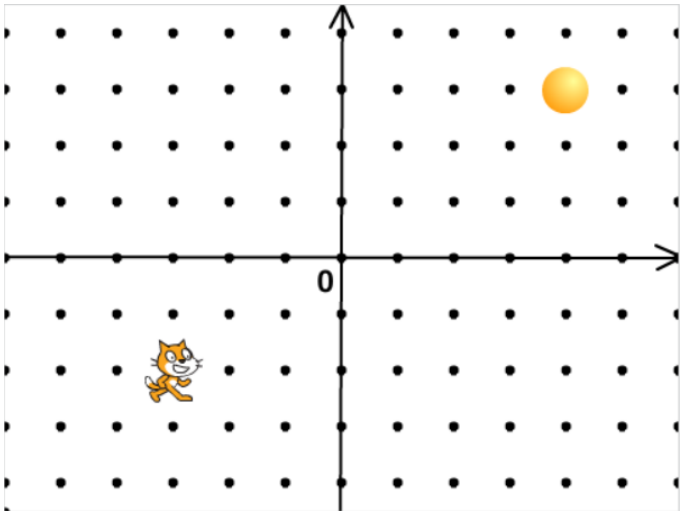
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	1
multiplicité des lutins	2
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	1
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	1
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	1
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

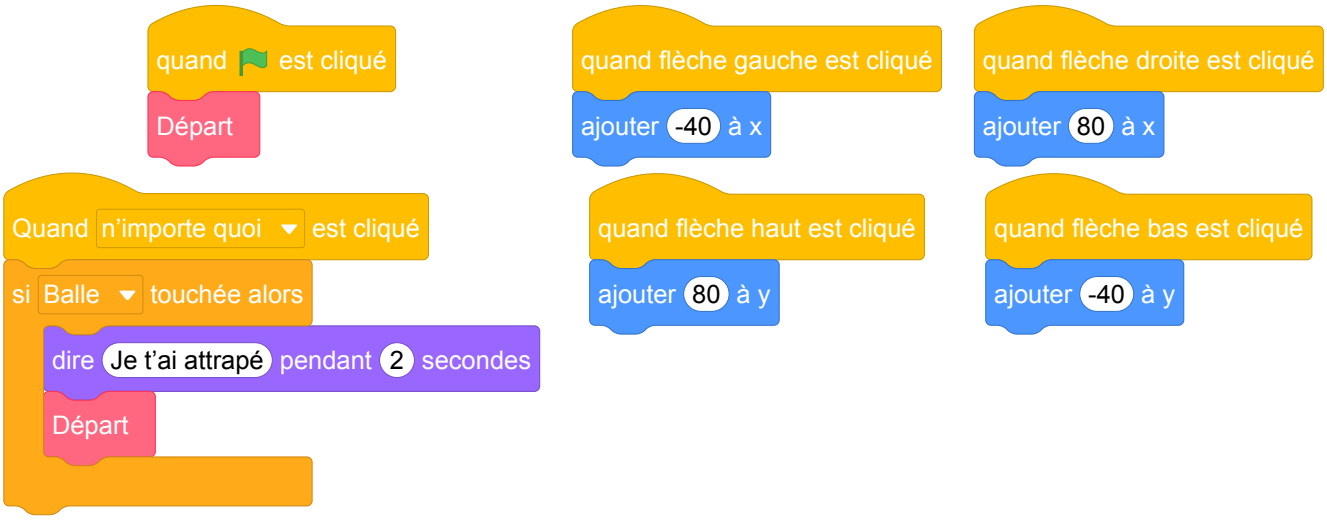
Énoncé - Amérique du nord, Juin 2017

L'image ci-dessous représente la position obtenue au déclenchement du bloc départ d'un programme de jeu.



L'arrière-plan est constitué de points espacés de 40 unités. Dans cette position, le chat a pour coordonnées (−120 ; −80).
Le but du jeu est de positionner le chat sur la balle.

1. Quelles sont les coordonnées du centre de la balle représentée dans cette position ?
2. Dans cette question, le chat est dans la position obtenue au déclenchement du bloc départ.
   Voici le script du lutin « chat » qui se déplace.



a. Expliquez pourquoi le chat ne revient pas à sa position de départ si le joueur appuie sur la touche → puis sur la touche ←.
b. Le joueur appuie sur la succession de touches suivante : → → ↑ ← ↓.
Quelles sont les coordonnées *x* et *y* du chat après ce déplacement ?
c. Parmi les propositions de succession de touches ci-dessous, laquelle permet au chat d'atteindre la balle ?

Déplacement 1	Déplacement 2	Déplacement 3
→→→→→→→↑↑↑↑↑	→→→↑↑↑→↓←	↑→↑→↑→→↓↓

3. Que se passe-t-il quand le chat atteint la balle ?

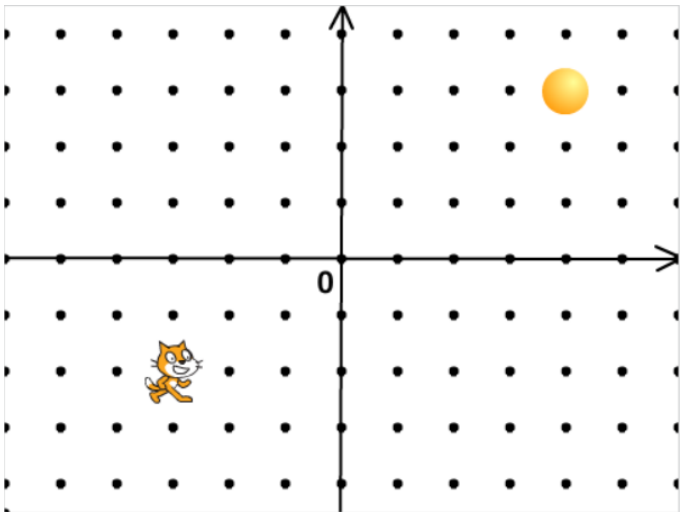
Correction

Fichier scratch : `sg_amerique_du_nord_2017_juin.sb3`

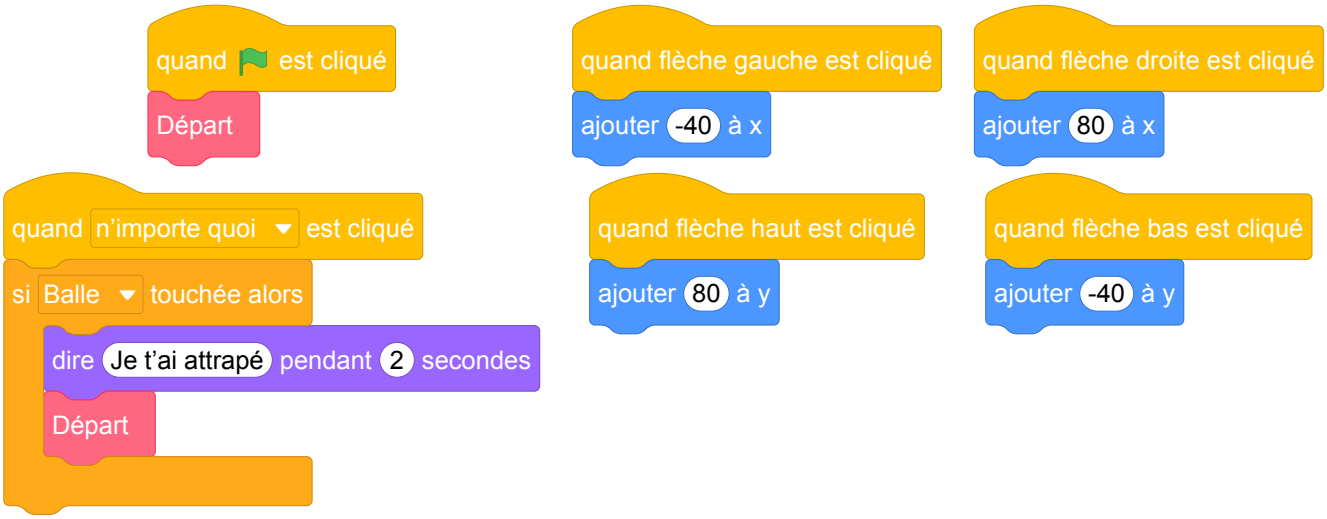
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/362195550/`

Solution proposée par l'APMEP

1. L'abscisse de la balle est 4×40. L'ordonnée de la balle est 3×40. Le centre de la balle a donc pour coordonnées $(160\,;\,120)$.



2. Voici le script du lutin « chat » qui se déplace.



a. Vers la droite il y a déplacement de 80 unités alors que vers la gauche on se déplace de 40 unités.

b. Horizontalement le déplacement est de : $2 \times 80 - 1 \times 40 = 160 - 40 = 120$ et verticalement : $1 \times 80 - 1 \times 40 = 80 - 40 = 40$.

Le chat est donc au point de coordonnées $(0\,;\,-40)$.

c. Parmi les propositions de succession de touches ci-dessous, laquelle permet au chat d'atteindre la balle ?

Déplacement 1	Déplacement 2	Déplacement 3
→→→→→→→↑↑↑↑↑	→→→↑↑↑→↓←	↑→↑→↑→→↓↓
$7 \times 80 = 560$ horizontalement $5 \times 80 = 400$ verticalement arrivée en $(440\,;\,320)$	$4 \times 80 - 1 \times 40 = 280$ horizontalement $3 \times 80 - 1 \times 40 = 200$ verticalement arrivée en $(160\,;\,120)$	$4 \times 80 = 320$ horizontalement $3 \times 80 - 2 \times 40 = 160$ verticalement arrivée en $(200\,;\,80)$

C'est donc le déplacement 2.

3. Quand le chat atteint la balle il s'affiche pendant 2 secondes : « Je t'ai attrapé ».

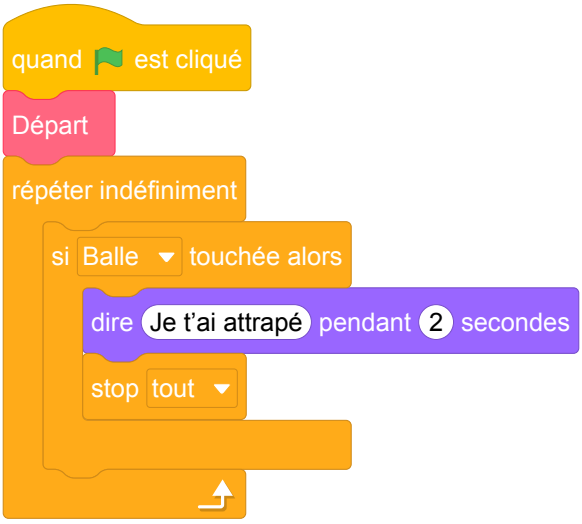
Analyse

Il s'agit d'un exercice mettant en jeu les aspects événementiels dans le cadre d'un jeu. La graduation du repère, avec échelle donnée permet à l'élève de bien se repérer. Néanmoins, l'exercice est très ambigu, fortement basé sur la connaissance de Scratch même si, curieusement, les blocs utilisés ne correspondent pas aux blocs fournis par Scratch : par exemple, lorsqu'il s'agit d'un événement déclenché par l'appui sur une touche, le vocabulaire utilisé par Scratch est bien "pressé" et non "cliqué".

Plus ennuyeux, l'exercice semble méconnaître plusieurs points importants au sujet de la programmation en parallèle. Dans un script, il faut à tout prix éviter le déclenchement de deux scripts par le même événement car on ne sait pas quel script s'exécute en premier. Concrètement, l'expérience montre qu'en Scratch lorsque deux scripts sont concurrents, c'est celui qui a été déplacé ou modifié en dernier qui s'exécute en premier (une analyse poussée de ces problèmes et des concepts sous-jacents est proposée par Anne Heam dans [5]).

Dans l'exercice, lorsque l'on presse une touche « flèche », il est impossible de savoir si le dernier script « n'importe quoi cliqué » s'exécute avant ou après le déplacement. Il existe donc une possibilité que le lutin dépasse la balle sans que la collision soit détectée. Il y a également un problème lié à la file d'attente des instructions. Selon que le déplacement a lieu avant ou après le test, quelle suite d'instructions est-elle exécutée ? Cet exercice formulé tel quel pose donc problème dans le cadre d'un examen, mais il peut être intéressant en classe, en cours de formation.

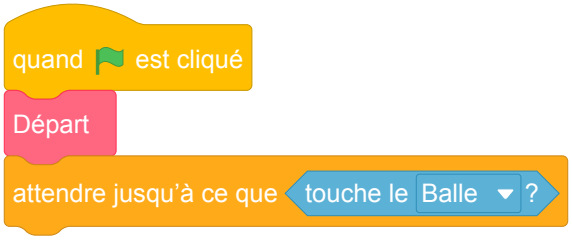
En outre, il est tout à fait possible d'imaginer un nouveau programme ayant le comportement que l'on attend. Pour éviter d'avoir deux scripts déclenchés par le même événement, on peut, par exemple, utiliser, dans le script déclenché par le drapeau vert, une boucle infinie contenant une instruction conditionnelle qui réalise le test d'intersection :



Fichier scratch : `sg_amerique_du_nord_2017_juin_v1.sb3`

Le programme ainsi conçu utilise un mécanisme appelé *attente active*, c'est à dire une boucle qui teste de manière répétée si une condition est vérifiée avant de laisser la suite du programme s'exécuter. C'est un peu comme si quelqu'un voulant entrer dans une pièce déjà occupée, frappait de manière répétée à la porte plutôt que d'attendre d'être prévenu par la personne lorsqu'elle libère la pièce. D'autre part, même si on peut supposer que le rythme de répétition de la boucle « répéter indéfiniment » est suffisant pour garantir un bon fonctionnement, cela n'est toujours pas garanti (par exemple si un utilisateur laisse une touche appuyée).

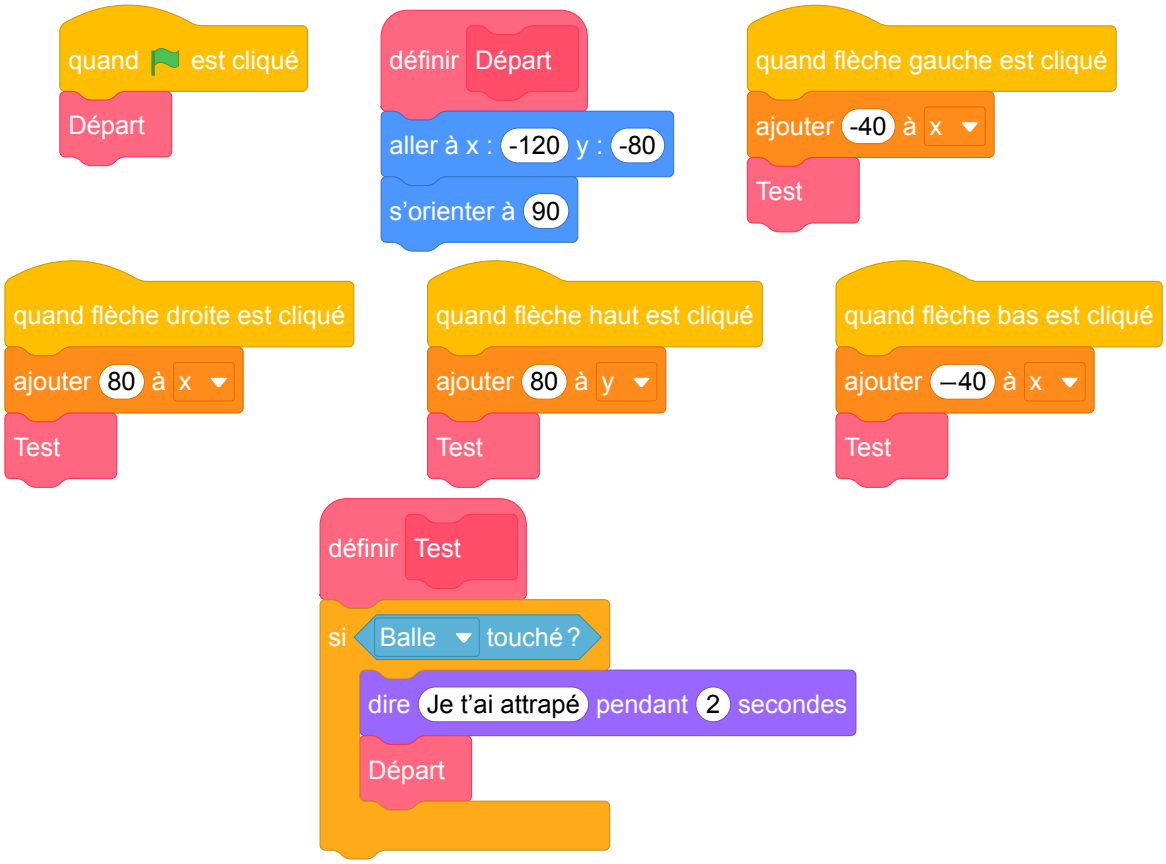
Une solution plus simple est d'utiliser le bloc « attendre » fourni par Scratch, qui permet de suspendre l'exécution d'un script jusqu'à ce qu'un certain événement se produise :



Fichier scratch : `sg_amerique_du_nord_2017_juin_v2.sb3`

Ce script est essentiellement équivalent à la solution précédente par attente active, la boucle infinie précédente étant ici implicitement prise en charge par l'environnement Scratch.

Une dernière solution consiste à inclure le test de collision à la fin de chacun des 4 blocs. Après tout, on n'a besoin de vérifier que le lutin touche la balle que lorsqu'il a réalisé un déplacement. Cela garantit également une uniformité de comportement, puisque le test n'est effectué à aucun autre moment. La proposition de programme ci-dessous suit cette approche plus cohérente, et permet de faire faire cet exercice sans changer beaucoup de questions :



Fichier scratch : `sg_amerique_du_nord_2017_juin_v3.sb3`

6 Asie, juin 2017, série générale

Références

Série : générale

Académie : Asie

Date : juin 2017

Exercice 4 - 4 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie sans transformation
- repérage
- déplacement

Thèmes informatiques

- scène
- activité du lutin

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	1
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	1
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Asie, Juin 2017

Margot a écrit le programme suivant. Il permet de dessiner avec trois touches du clavier.

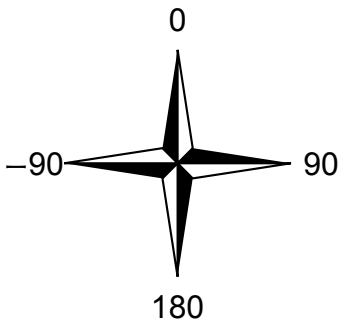
```
quand [drapeau] est cliqué
initialisation

quand flèche haut est cliqué
s'orienter à 0
stylo en position d'écriture
avancer de 50 pas
relever le stylo

quand flèche droite est cliqué
s'orienter à 90
stylo en position d'écriture
avancer de 50 pas
relever le stylo

quand flèche bas est cliqué
s'orienter à 180
stylo en position d'écriture
avancer de 50 pas
relever le stylo
```

Pour information

```
initialisation
```

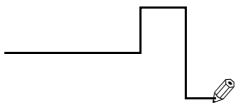
Ce bloc efface le dessin précédent, positionne le crayon à gauche de l'écran et relève le stylo.

```
s'orienter à 90
90 à droite
-90 à gauche
0 vers le haut
180 vers le bas
```

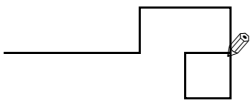


1. Parmi les trois dessins suivants, un seul ne pourra pas être réalisé avec ce programme. Lequel ? Expliquer.

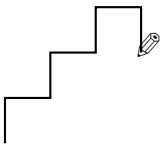
Dessin 1



Dessin 2



Dessin 3



2. Julie a modifié le programme de Margot (voir ci-dessous). Que devient alors le dessin 3 avec le programme modifié par Julie ?

Programme modifié par Julie

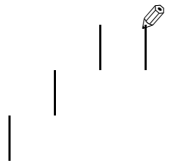
```
quand [drapeau] est cliqué
initialisation

quand flèche haut est cliqué
s'orienter à 0
stylo en position d'écriture
avancer de 50 pas
relever le stylo

quand flèche droite est cliqué
s'orienter à 90
avancer de 50 pas

quand flèche bas est cliqué
s'orienter à 180
stylo en position d'écriture
avancer de 50 pas
relever le stylo
```

Correction

Fichier scratch : `sg_asie_2017_juin.sb3`

Lien vers le fichier Scratch en ligne : `https://scratch.mit.edu/projects/362202237/`

1. Le dessin 2 ne peut être réalisé car il n'y a pas d'instruction pour aller à gauche.
2. Le programme modifié fait la même chose sauf lorsqu'on appui sur la flèche droite : le déplacement a bien lieu mais le stylo n'est plus posé et donc aucun trait n'est tracé. Ainsi on obtient la même figure sans les traits horizontaux qui provenaient tous de l'appui sur cette touche.
   Le dessin devient alors :



Analyse

Les compétences évaluées par ce problème semblent être la compréhension de la programmation événementielle, la lecture d'un programme Scratch et la capacité à exécuter un programme pas à pas. L'énoncé comporte quelques erreurs et maladresses de formulation. Le bloc « initialisation » ne fait pas ce qu'il devrait faire puisque le lutin est déplacé avant de lever le stylo, un trait en trop va donc être tracé entre la position du crayon avant initialisation et la position "à gauche de l'écran". Par ailleurs la position du stylo sur l'axe des ordonnées après initialisation n'est pas spécifiée.

Dans la question 2, les modifications apportées par Julie au programme de Margot ne sont pas explicitées. Pour répondre, les élèves doivent donc d'abord les repérer. Or, le but de l'exercice n'est pas de trouver les modifications mais de comprendre ce qu'elles font. Aussi, serait-il préférable d'indiquer explicitement dans le texte où ont eu lieu les changements, afin de ne pas ajouter, pour certains élèves, une difficulté supplémentaire sans rapport avec les objectifs du sujet.

Cette question contient également quelques problèmes de formulation. En annonçant dans la question que le dessin 3 change, on donne un indice sur la question précédente, en excluant de fait ce dessin. De plus, le dessin 3 ne change pas, puisqu'il est le produit de la séquence d'instructions précédente. Julie crée une nouvelle séquence d'instructions qui donne une nouvelle figure.

Enfin, il pourrait être intéressant d'ajouter une question intermédiaire entre les questions 1 et 2, formulée par exemple ainsi : « Pour les deux autres dessins, quelle est la séquence de touches pressées par l'utilisateur ? ». La question 2 pourrait alors devenir : « Julie a modifié le programme de Margot (voir ci-dessous). Que dessine le lutin avec le programme modifié par Julie pour les séquences de touches trouvées à la question précédente ? ».

Propositions

Jeu « tout blanc tout noir » On peut proposer, pour travailler sur les événements, un travail autour du jeu « tout blanc tout noir ». Ce jeu se définit comme suit :

— un tableau carré contient des cases, 9 au départ par exemple ;

— chaque case peut être de deux couleurs, blanche ou noire ;

— lorsque l'on clique sur une case, la case cliquée et les cases adjacentes horizontalement et verticalement changent de couleur ;

— le but du jeu est, à partir d'une situation initiale, par exemple un tableau tout blanc, d'arriver à un tableau entièrement noir.

Dans Scratch, on peut créer des lutins, chacun représentant une case, avec deux costumes (l'un noir, l'autre blanc). On duplique ce lutin autant de fois que nécessaire. Il reste à programmer chaque lutin pour qu'il occupe la place d'une des cases du tableau. Pour la suite, on considérera que les cases sont nuérotées ainsi :

1	2	3
4	5	6
7	8	9

Pour programmer un tel jeu avec Scratch, on utilisera le système de passage de messages entre lutins. Lorsqu'on clique sur une case, un message est envoyé par le lutin correspondant. On considérera ici que le fait de cliquer sur la case *n* (et donc sur le lutin qui s'y trouve) envoie le message *n*.

En Scratch, un message envoyé par un lutin peut être reçu par tous les autres, mais seuls les lutins des cases voisines à la case *n* seront programmés pour réagir au message *n* en changeant de couleur (c'est à dire en passant du blanc au noir et inversement). Par exemple, si depuis l'état initial du jeu (toutes cases blanches) on clique sur le lutin (et donc la case) 2, les lutins des cases 1, 3 et 5 réagissent au message et changent de couleur. On obtient donc la figure suivante :

1	2	3
4	5	6
7	8	9

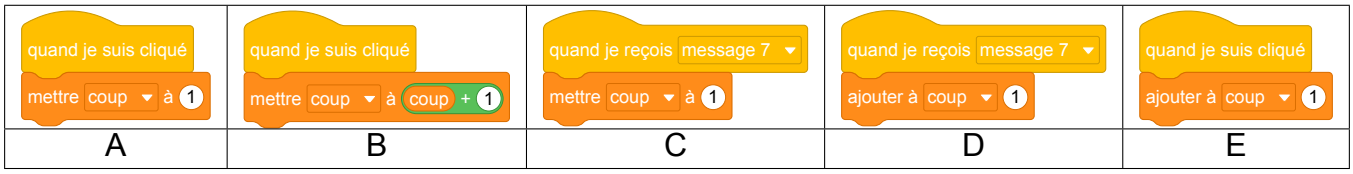
Si on clique ensuite sur le lutin (donc la case) 6, le message 6 est envoyé, les lutins des cases 3, 5 et 9 réagissent au message et on obtient la figure suivante :

1	2	3
4	5	6
7	8	9

Voici un exemple de questions que l'on pourrait poser aux élèves au sujet de ce jeu. On note que tout cet exercice suppose une certaine connaissance de la sémantique propre du langage Scratch.

1. Quelles sont les cases que l'on peut cliquer pour faire changer la case 7 de couleur ?
2. Pour tenir compte d'un clic sur la case 1, quel programme est approprié pour le lutin de la case 2 ?

3. Combien de scripts de changement de costume le lutin 5 possède-t-il (ne pas oublier qu'une case doit changer de couleur quand on clique dessus) ?
4. Donner les numéros des messages qui font changer la case 5 de couleur.
5. On souhaite ajouter un compteur de coups, pour savoir qui réussit le jeu le plus rapidement. On crée la variable coup pour tous les lutins. Quel(s) programme(s) associé(s) au lutin 7 parmi les suivants permet(tent) de rajouter 1 au compteur de coup quand la case 7 est cliquée ?



Réponses aux questions

1. On peut cliquer sur les cases 4, 7 et 8 pour changer la couleur de la case 7.
2. C'est le programme D qui convient. En effet, dans le programme B, la commande affecte un costume spécifique au lutin, et il ne pourra pas en changer au prochain clic. Les programmes A et C ne réagissent pas au message 1.
3. Le lutin 5 est associé à 5 changements de costume. Il ne faut pas oublier qu'il change de couleur lorsque l'on clique sur lui-même.
4. Parmi les messages qui font changer le lutin 5 on trouve les messages émis par les lutins adjacents, c'est à dire 2, 4, 6 et 8. Quand un lutin envoie un message, tous les lutins le reçoivent, même celui qui envoie le message. Donc quand on clique sur le lutin 5 ce qui déclence l'envoi du message 5, on a le choix de le faire changer de costume à la réception du message 5 ou bien d'inclure un script qui réagit au clic de ce lutin pour le changement de costume.
5. Les séquences A et C initialisent la variable à 1 systématiquement et ne conviennent donc pas. Les séquences B et E permettent de rajouter une unité au compteur de coup puisqu'elle sont placée dans le script du lutin 7. La séquences D peut fonctionner en étant associée à n'importe quel Lutin.

Lien vers le fichier Scratch en ligne : `https://scratch.mit.edu/projects/772294819/`

7 Centres étrangers, juin 2017, série générale

Références

Série : générale générale

Académie : Centres étrangers

Date : juin 2017

Exercice 6 – 9 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- frise
- grandeurs et mesures

Thèmes informatiques

- itération bornée
- bloc sans paramètre

Compétences travaillées

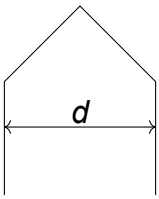
- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	1
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Centres étrangers, Juin 2017

Pour tracer une « rue », on a défini le tracé d'une « maison ».

```
définir maison
tourner ↺ de 90 degrés
avancer de 50 pas
tourner ↻ de 45 degrés
avancer de 50 pas
tourner ↻ de 90 degrés
avancer de 50 pas
tourner ↻ de 45 degrés
avancer de 50 pas
tourner ↺ de 90 degrés
```



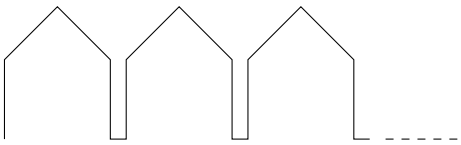
Tracé de la « maison »

```
quand 🚩 est cliqué
cacher
mettre la taille du stylo à 1
aller à x : -240 y : 0
effacer tout
stylo en position écriture
s'orienter de 90
répéter n fois
    maison
    avancer de 20 pas
```

Programme principal

1. Vérifier que d est environ égal à 71 à l'unité près.
2. Un point dans une fenêtre d'exécution de votre programme a son abscisse qui peut varier de -240 à 240 et son ordonnée qui peut varier de -180 à 180.

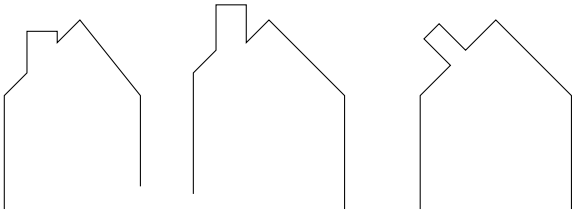
   Quel est le plus grand nombre entier n que l'on peut utiliser dans le programme principal pour que le tracé de la « rue » tienne dans la fenêtre de votre ordinateur où s'exécute le programme ?

   Vous pourrez tracer sur votre copie tous les schémas (à main levée ou non) qui auront permis de répondre à la question précédente et ajouter toutes les informations utiles (valeurs, codages, traits supplémentaires, noms de points…)
3. *Attention, cette question est indépendante des questions précédentes et la « maison » est légèrement différente.*

   Si on désire rajouter une sortie de cheminée au tracé de la maison pour la rendre plus réaliste, il faut faire un minimum de calculs pour ne pas avoir de surprises.

   Exemples :

On suppose que :

— les points H, E et A sont alignés ;

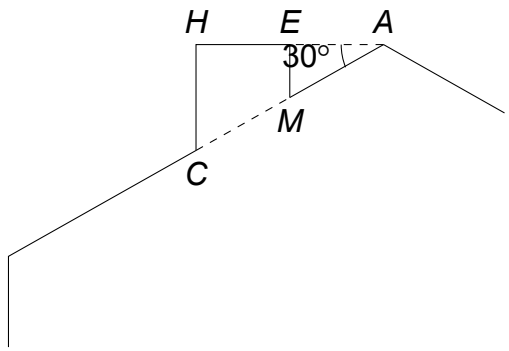
— les points C, M et A sont alignés ;

— $[CH]$ et $[EM]$ sont perpendiculaires à $[HA]$;

— $AM = 16$;

— $MC = 10$;

— $\widehat{HAC} = 30°$.



Calculer EM, HC et HE afin de pouvoir obtenir une belle sortie de cheminée.

Correction

Fichier scratch : `sg_centres_etrangers_2017_juin.sb3`

Lien vers le fichier Scratch en ligne : `https://scratch.mit.edu/projects/772309312`

1. Le sommet de la maison est un triangle rectangle d'hypoténuse d et dont les autres cotés mesurent 50 unités. D'après le théorème de Pythagore on a donc :
   $d^2 = 50^2 + 50^2 = 2500 + 2500 = 5000$, donc $d = \sqrt{5000} \approx 70,7$ soit 71 unités à l'unité prés.
2. Chaque motif (maison plus avancée de 20 unités) prend horizontalement environ 91 unités.
   Or $5 \times 91 = 459$ et $6 \times 91 = 546$.
   On peut donc démarrer à -240 et dessiner 5 motifs soit 5 maisons.
3. Dans le triangle AEM rectangle en E, on a $\sin \widehat{EAM} = \dfrac{EM}{AM}$, soit $\dfrac{1}{2} = \dfrac{EM}{16}$ soit $EM = 16 \times \dfrac{1}{2} = 8$.
   De la même façon, dans le triangle AHC rectangle en H, $\dfrac{1}{2} = \dfrac{HC}{16+10}$ soit $HC = 26 \times \dfrac{1}{2} = 13$.
   D'autre part $AE = AM \times \cos 30 \approx 13,86$ et $AH = AC \times \cos 30 \approx 22,52$, donc $HE = AH - AE \approx 22,52 - 13,86$, donc $HE \approx 8,66$.

Analyse

— Dans la question 1, du point de vue mathématique, la figure n'est pas considérée pour son orientation et c'est le sens de cette première question. Cependant du point de vue informatique, la figure produite est dépendante de l'état du lutin.

  Il aurait été sans doute pertinent de scinder cette question en trois : une première question portant sur l'état initial du lutin, que l'on peut déduire du programme principal, une deuxième demandant de dessiner le tracé produit par l'exécution du bloc « maison » sur une annexe comportant un repère et un état initial particulier, et une troisième portant sur le calcul de la distance d.

  Le traitement de la première question relative à l'état initial du lutin relève d'une connaissance du langage Scratch puisque

signifie que le lutin « regarde » vers la droite de la scène.

  Ce fait est d'ailleurs rappelé par le rédacteur dans d'autres sujets analysés dans ce texte.

— La question 2 fait appel à des connaissances à la fois informatiques et mathématiques, puisque la question « quel est le plus grand entier n… » nécessite d'avoir compris le fonctionnement d'une boucle. Ici, le contexte est spécifié dans le programme principal (position, direction et visibilité du lutin, taille et position du stylo).

  Toutefois une clarification est nécessaire sur le fait que la dernière maison tracée doive être complètement dessinée. Une reformulation possible est « Quel est le nombre maximal de maisons complètes qui forment la rue dans la scène ? ».

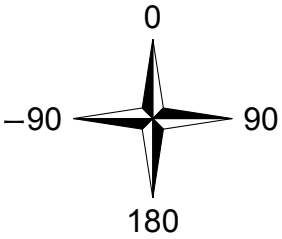
— Concernant la question 3, il nous semblerait intéressant de proposer un bloc « cheminée » permettant de tracer la partie du toit sur laquelle repose la cheminée, et de demander quelles modifications du bloc maison permettent ainsi de produire la maison avec une cheminée.

  Nous avons défini un bloc cheminée.

Propositions

Nous proposons ci-dessous une ébauche de question.

> Les instructions ci-dessous permettent d'obtenir la figure de du milieu, qui représente le pan complet du « toit » sur lequel se trouve la cheminée. La rose des vents qui se trouve à droite rapelle les orientations possible du lutin : 90 oriente vers la droite, 0 vers le haut ...
>
> Quelles modifications de la définition du bloc maison permettent de produire la maison complète ?



L'adaptation proposée fait alors apparaître un découpage en sous-problèmes, et permet de questionner la capacité de l'élève à ordonner l'exécution des instructions du programme et l'état de la variable orientation du lutin.

8 Métropole – La Réunion – Antilles – Guyane, juin 2017, série générale

Références

Série : générale générale

Académie : Métropole – La Réunion – Antilles – Guyane

Date : juin 2017

Exercice 2 - 6 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- repérage
- spirale

Thèmes informatiques

- variable déclarée
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

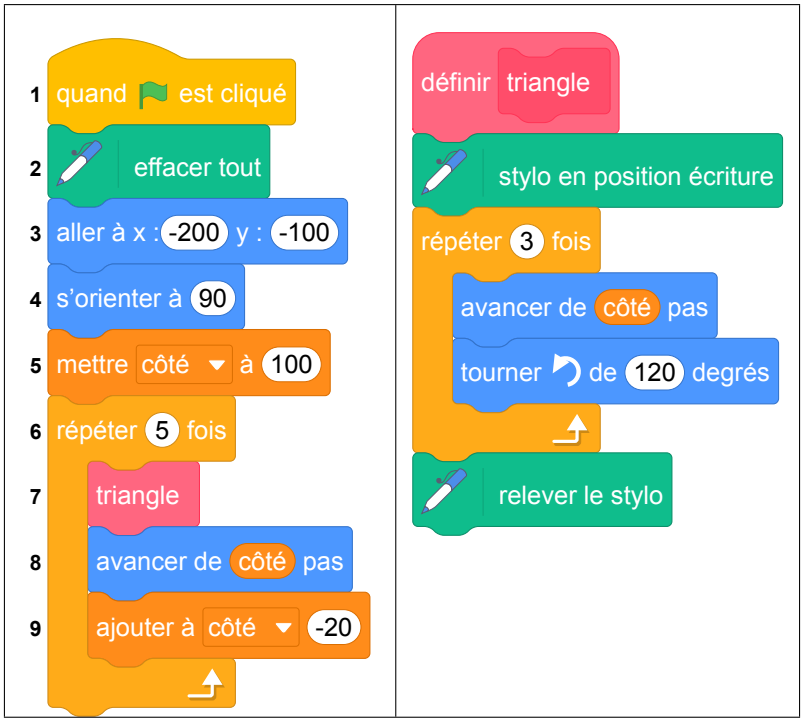
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion – Antilles – Guyane, Juin 2017

On donne le programme suivant qui permet de tracer plusieurs triangles équilatéraux de tailles différentes.

Ce programme comporte une variable nommée « **côté** ». Les longueurs sont données en pixels.

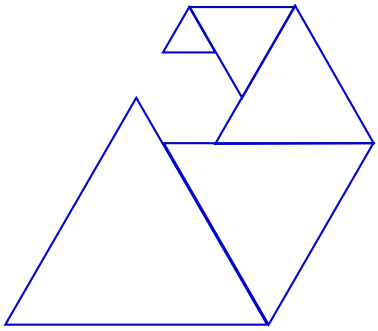
On rappelle que l'instruction [s'orienter à 90] signifie que l'on se dirige vers la droite.



1. Quelles sont les coordonnées du point de départ du tracé ?
2. Combien de triangles sont dessinés par le script ?
3. (a) Quelle est la longueur (en pixels) du côté du deuxième triangle tracé ?

   (b) Tracer à main levée l'allure de la figure obtenue quand on exécute ce script.
4. On modifie le script initial pour obtenir la figure ci-contre.

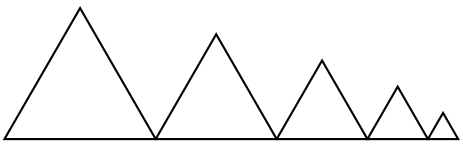
   Indiquer le numéro d'une instruction du script **après laquelle** on peut placer l'instruction [tourner de 60 degrés] pour obtenir cette nouvelle figure.



Correction

Fichier scratch : `sg_metropole_2017_juin.sb3`
Lien vers le fichier Scratch en ligne : `https://scratch.mit.edu/projects/362204586`

1. Les coordonnées du point de départ du tracé sont $(-200\ ;\ -100)$.
2. Le script permet de dessiner 5 triangles.
3. (a) La longueur du côté du deuxième triangle tracé est de 80 pixels.
   (b) La figure obtenue :



4. Il faut placer le bloc « tournez le bloc de 60° » après l'instruction n° 9 du script initial pour obtenir cette nouvelle figure.

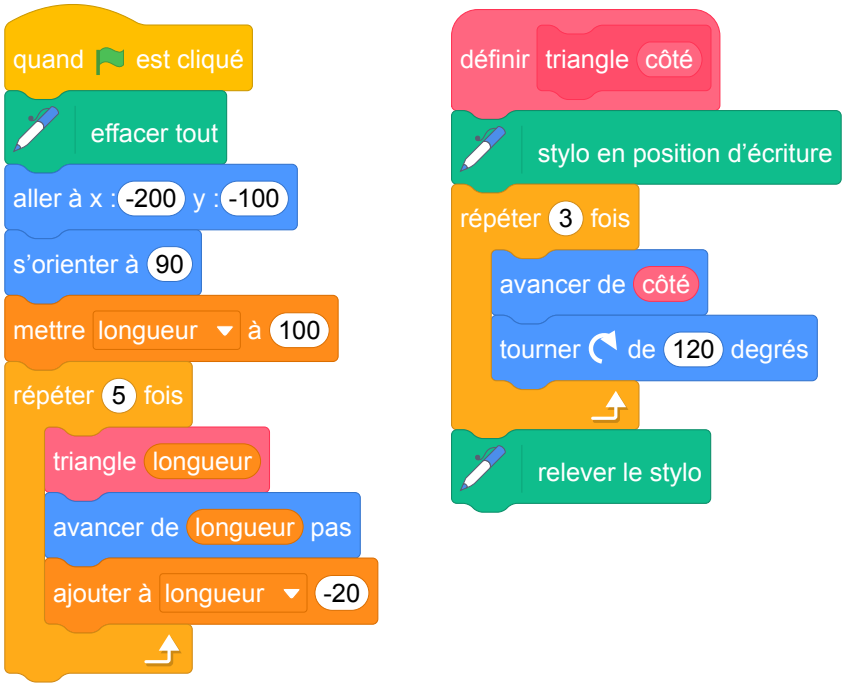
Analyse

L'exercice semble comporter un saut de difficulté important à la question 4.

Il reste cependant le problème de la variable « côté » , qui est globale et qui est utilisée dans le bloc (procédure) « triangle ». Il serait plus judicieux de proposer un bloc à paramètre, dans ce programme.

Propositions

Pour éviter l'écueil d'une variable globale, il est possible d'utiliser un bloc avec un paramètre comme illustré dans le programme ci-dessous.



9 Métropole, La Réunion, Antilles-Guyanne, septembre 2017, série générale

Références

Série : générale

Académie : Métropole, La Réunion, Antilles, Guyanne

Date : septembre 2017

Exercice 3 - 6,5 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie sans transformation
- spirale

Thèmes informatiques

- variable réservée
- variable déclarée
- activité du lutin
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	1
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion – Antilles-Guyane, 14 septembre 2017

Énoncé (exercice 3, 6 points sur 50)

Voici trois figures différentes, aucune n'est à l'échelle indiquée dans l'exercice :

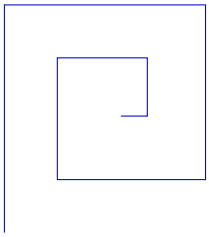


figure 1

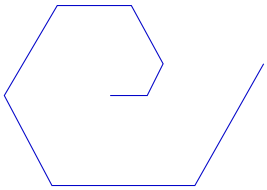


figure 2

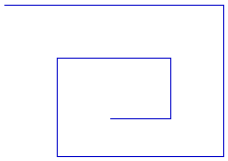
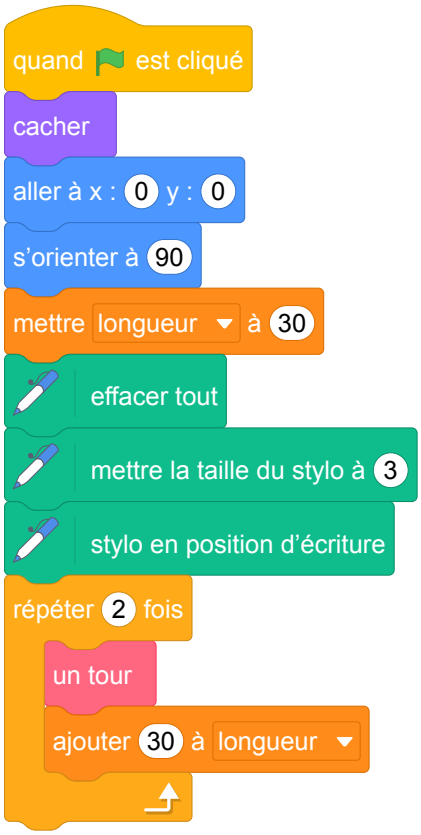


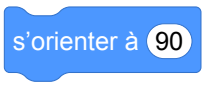
figure 3

Le programme ci-dessous contient une variable nommée « **longueur** ».

Script



Le bloc : **un tour**

On rappelle que l'instruction 

signifie que l'on s'oriente vers la droite avec le stylo.

1. (a) Dessiner la figure obtenue avec le bloc « un tour » donné dans le cadre de droite ci-dessus, pour une longueur de départ égale à 30, étant orienté vers la droite avec le stylo, en début de tracé. On prendra 1 cm pour 30 unités de longueur, c'est-à-dire 30 pas.

   (b) Comment est-on orienté avec le stylo après ce tracé ? Aucune justification n'est demandée.

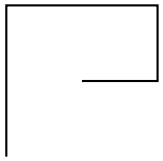
2. Laquelle des figures 1 ou 3 le programme ci-dessus permet-il d'obtenir ? Justifier votre réponse.
3. Quelle modification faut-il apporter au bloc « **un tour** » pour obtenir la figure 2 ci-dessus ?

Correction

Fichier scratch : `sg_metropole_la_reunion_antilles_guyanne_2017_septembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/362216866/`

1. (a)



   (b) On a tourné quatre fois de 90°, donc fait un tour : le stylo est encore orienté vers la droite.

2. Ce ne peut être la figure 1 puisque l'on déplace de 30 puis de 60, alors que dans le tour on répète deux déplacements de 30.

   Ce ne peut être la figure 2 puisque l'on tourne après chaque déplacement de 60°.

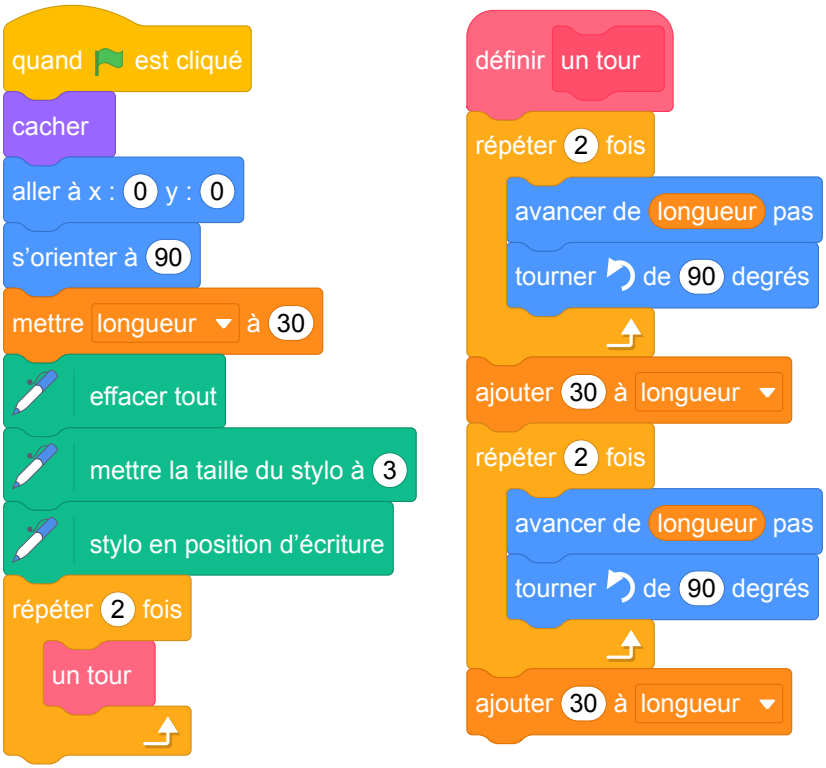
   Il ne reste donc que la figure 3.

3. Les déplacements augmentent bien de longueur à chaque fois ; il suffit donc de tourner de 60° pour obtenir la figure 2.

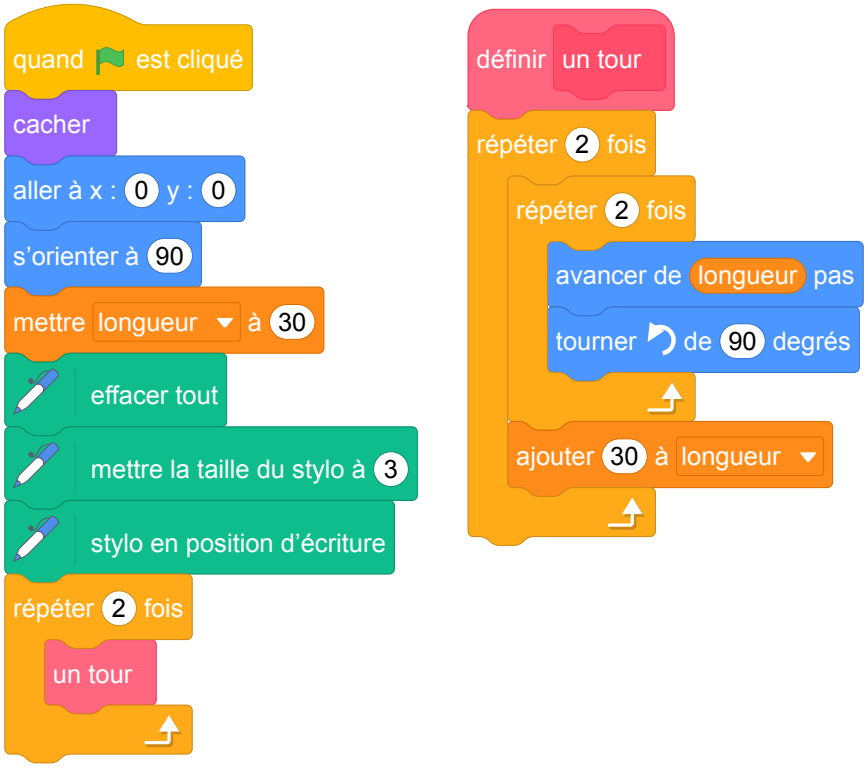
Analyse

Cet exercice assez riche comporte en particulier des questions d'interprétation de code. On note une bonne précision générale et globalement une bonne rédaction de l'énoncé (exception faite du problème des unités de longueur précédemment mentionné). La prise en compte de l'état "caché" (position, orientation) est intéressante. L'exercice porte à la fois sur des compétences mathématiques et informatiques. Signalons cependant deux maladresses d'expression dans la question 1(b) : on ne sait pas qui représente le « on » et l'expression « orientation du stylo » n'est pas appropriée, puisque c'est le lutin qui est orienté.

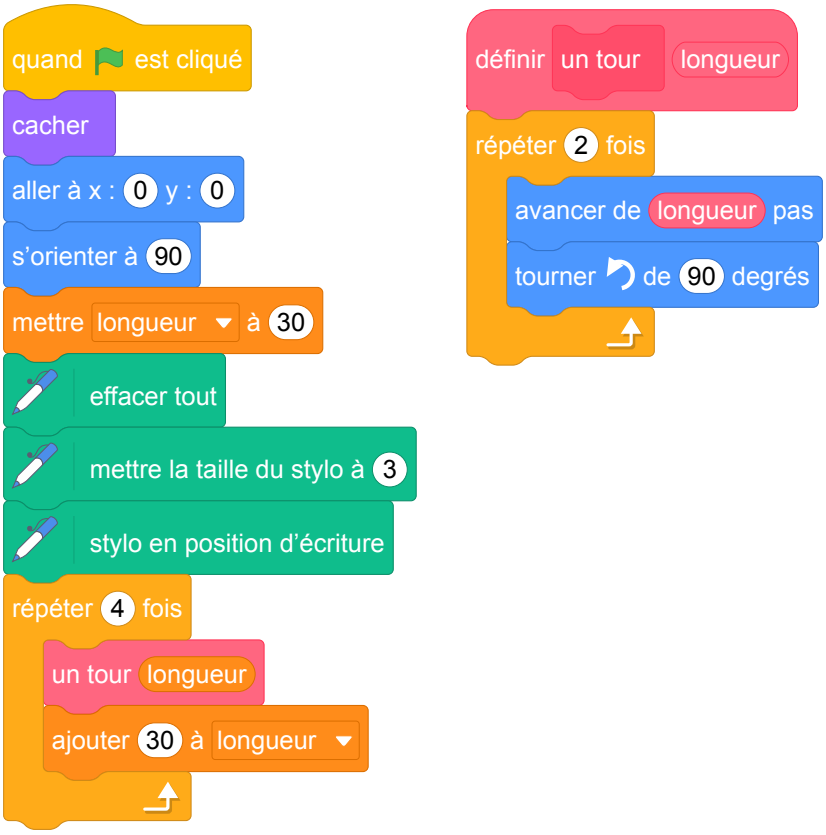
Version 1



Version 2



Version 3



Quant au programme lui-même, il pourrait être aussi quelque peu amélioré. Nous proposons donc 3 versions modifiées de ce programme (détaillées ci-dessus). Le problème de la variable globale déjà mentionné est réglé dans la version 3 via l'utilisation d'un bloc avec un paramètre. Notons également que le nom du bloc "un tour" n'est peut-être pas des plus pertinents surtout lorsqu'il s'agit de modifier ledit bloc pour obtenir la figure 2. Le nommer « motif » présenterait peut-être moins de risque de confusion. D'autre part, il est un peu curieux de modifier la valeur de la variable longueur une fois dans le bloc et une fois dans la boucle. Il serait plus cohérent de réaliser les deux mises à jour de la variable dans le bloc lui-même (Version 1). Ceci fait, le bloc pourrait alors être écrit de manière un peu plus concise (cf Version 2). On aurait aussi pu définir un motif de base ne comportant que 2 côtés au lieu de 4 et construire la figure en répétant ce motif 4 fois au lieu de 2 (cf Version 3).

Nous insistons sur le fait que les blocs de Scratch ne sont pas des fonctions, puisqu'on ne peut pas renvoyer une valeur précise, mais plutôt des procédures, ce qui en fait un système peu pratique. Enfin, le paramètre en entrée, comme dans la version 3, n'est pas une variable au sens strict, puisque sa valeur ne peut pas être modifiée à l'intérieur du bloc, mais seulement être utilisée par les autres instructions.

10 Amérique du Sud, novembre 2017, série générale

Références

Série : générale

Académie : Amérique du Sud

Date : novembre 2017

Exercice 6 - 6,5 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- agrandissement reduction
- caractérisation des figures
- déplacement
- homothétie

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Thèmes informatiques

- variable_declaree
- iteration_bornee
- bloc_sans_parametres

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Amérique du sud, Novembre 2017

Le bloc d'instruction « carré » ci-dessous a été programmé puis utilisé dans les deux programmes ci-contre :

```
définir carré
stylo en position écriture
répéter 4 fois
    avancer de longueur pas
    tourner ↺ de 90 degrés
relever le stylo
```

Rappel :

L'instruction `avancer de 10 pas` fait avancer le lutin de 10 pas.

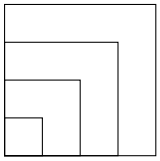
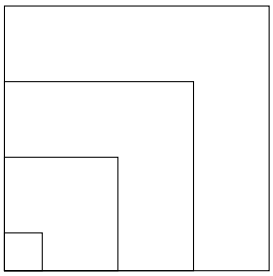
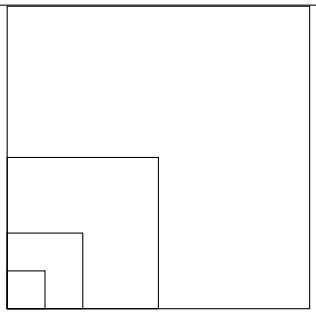
Programme n° 1

```
quand ⚑ est pressé
mettre longueur à 10
répéter 4 fois
    carré
    mettre longueur à longueur + 20
cacher
```

Programme n° 2

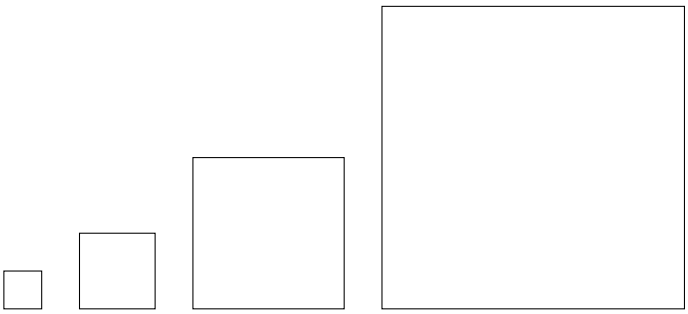
```
quand ⚑ est pressé
mettre longueur à 10
répéter 4 fois
    carré
    mettre longueur à longueur * 2
cacher
```

1. Voici trois dessins :

Dessin n° 1	Dessin n° 2	Dessin n° 3
		

   (a) Lequel de ces trois dessins obtient-on avec le programme n° 1 ?
   (b) Lequel de ces trois dessins obtient-on avec le programme n° 2 ?
   (c) Pour chacun des deux programmes, déterminer la longueur, en pas, du côté du plus grand carré dessiné ?

2. On souhaite modifier le programme n° 2 pour obtenir le dessin ci-dessous.



Parmi les trois modifications suivantes, laquelle permet d'obtenir le dessin souhaité ? Aucune justification n'est attendue pour cette question.

Modification 1	Modification 2	Modification 3
quand est pressé mettre longueur à 10 répéter 4 fois carré avancer de (longueur + 10) pas mettre longueur à (longueur * 2) cacher	quand est pressé mettre longueur à 10 répéter 4 fois carré mettre longueur à (longueur * 2) avancer de (longueur + 10) pas cacher	quand est pressé mettre longueur à 10 répéter 4 fois carré mettre longueur à (longueur * 2) avancer de (longueur + 10) pas cacher

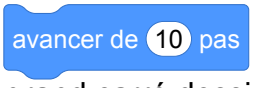
Correction

Fichier scratch : `sg_amerique_du_sud_2017_novembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/362210852/`

Solution proposée par l'APMEP

1. (a) Le dessin n° 2 est obtenu avec le programme n° 1.
   (b) Le dessin n° 3 est obtenu avec le programme n° 2.
   (c) Pour le programme n° 1 :
   — Le premier carré a une longueur de côté de 10 ;
   — le deuxième carré a une longueur de côté de 30, $(10 + 20)$;
   — le troisième carré a une longueur de côté de 50, $(30 + 20)$;
   — le quatrième carré a une longueur de côté de 70, $(50 + 20)$.

   L'instruction 

   fait avancer le lutin de 10 pas, donc la longueur, en pas, du côté du plus grand carré dessiné est égale à 70 pas.
   Pour le programme n° 2 : les dimensions du carré sont à chaque fois doublées ; la longueur, en pas, du côté du plus grand carré dessiné est égale à 80 pas.
2. La modification 1 permet d'obtenir le dessin souhaité.

Analyse

La compréhension d'algorithme est intéressante. Mais se limiter à ce type d'exercice, ne permet pas d'évaluer toutes les compétences attendues en fin de cycle 4. Il est ainsi dommage qu'il ne soit pas demandé de réaliser un dessin, ou a minima de modifier un des programmes proposés pour obtenir un dessin donné. Il faudrait regarder la longueur globale du sujet de mathématiques pour évaluer la possibilité de compléter cet exercice dans ce sens.

Propositions

Une fois les défauts évoqués précédemment corrigés, ces exercices sont tout à fait pertinents dans le cadre du brevet. Nous ne proposons donc pas ici de nouvel exercice mais recommandons, dans cette catégorie de problèmes, de combiner des questions de compréhension et analyse de programmes avec des exercices de conception d'algorithmes de dessin éventuellement par modification d'un programme donné.

Il est possible d'étendre l'exercice avec des questions complémentaires :

3. Exécuter un programme
   (a) Dessiner ce qui s'affiche à l'écran si on exécute le programme suivant. On prendra 5 mm pour représenter un pas.

Programme n° 3

(b) Quelle est la nature de la transformation qui permet de passer d'un carré au suivant ?

L'énoncé est également adaptable avec d'autres polygones, des triangles par exemple.

Énoncé modifié

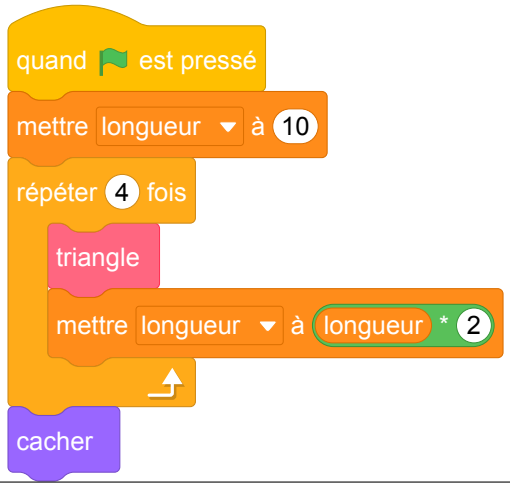
Le bloc d'instruction « triangle » ci-dessous a été programmé puis utilisé dans les deux programmes ci-contre :

Rappel :

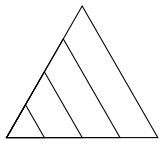
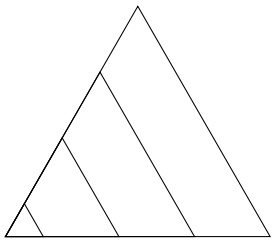
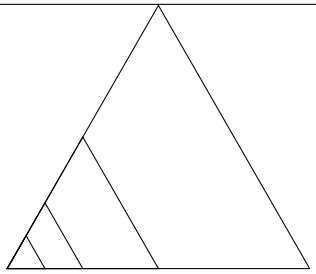
L'instruction avancer de 10 pas fait avancer le lutin de 10 unités de longueur Scratch (pixels).

Programme n° 1

Programme n° 2



1. Voici trois dessins :

Dessin n° 1	Dessin n° 2	Dessin n° 3
		

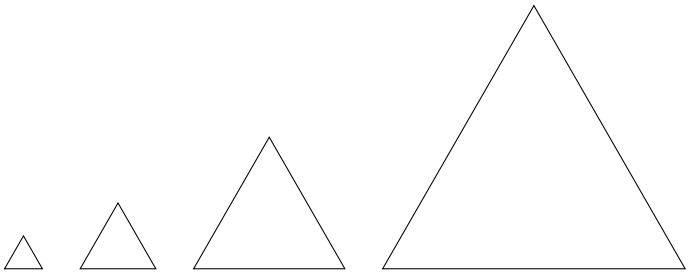
(a) Lequel de ces trois dessins obtient-on avec le programme n° 1 ?

(b) Lequel de ces trois dessins obtient-on avec le programme n° 2 ?

(c) Pour chacun des deux programmes, déterminer la longueur, en pas, du côté du plus grand triangle dessiné ?

(d) Dans programme n° 2, quelle est la nature de la transformation qui permet de passer d'un triangle au suivant ?

2. On souhaite modifier le programme n° 2 pour obtenir le dessin ci-dessous.



Parmi les trois modifications suivantes, laquelle permet d'obtenir le dessin souhaité ? Aucune justification n'est attendue pour cette question.

Modification 1	Modification 2	Modification 3

3. Exécuter un programme

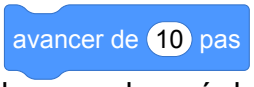
(a) Dessiner ce qui s'affiche à l'écran si on exécute le programme suivant. On prendra 5 mm pour représenter un pas.

Programme n° 3

(b) Quelle est la nature de la transformation qui permet de passer d'un triangle au suivant ?

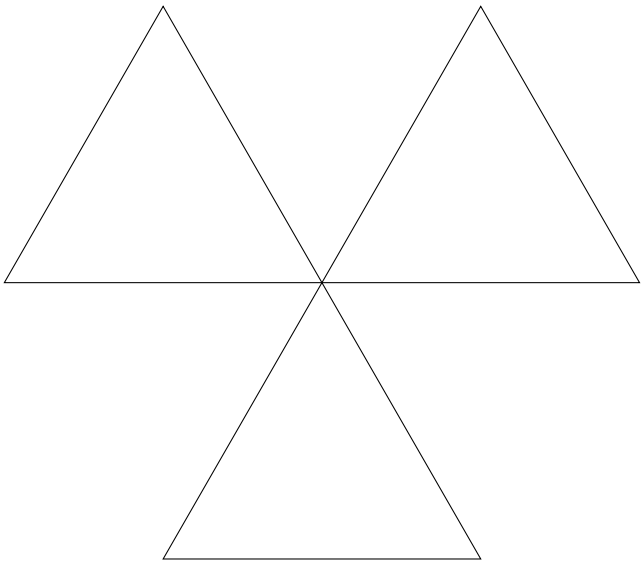
Correction de l'énoncé modifié

1.
   (a) Le dessin n° 2 est obtenu avec le programme n° 1.
   (b) Le dessin n° 3 est obtenu avec le programme n° 2.
   (c) — Pour le programme n° 1 :
      — Le premier triangle une longueur de côté de 10 ;
      — le deuxième triangle une longueur de côté de 30, $(10 + 20)$;
      — le troisième triangle une longueur de côté de 50, $(30 + 20)$;
      — le quatrième triangle une longueur de côté de 70, $(50 + 20)$.

      L'instruction 

fait avancer le lutin de 10 pas, donc la longueur, en pas, du côté du plus grand carré dessiné est égale à 70 pas.

   — Pour le programme n° 2 : les dimensions du triangle sont à chaque fois doublées ; la longueur, en pas, du côté du plus grand triangle dessiné est égale à 80 pas.
   (d) Dans programme n° 2, la transformation qui permet de passer d'un triangle au suivant est une homothétie de centre le sommet commun à tous les triangles et de rapport 2.
2. La modification 1 permet d'obtenir le dessin souhaité.
3. Exécuter un programme
   (a)



Chaque côté mesure 4 cm

   (b) La nature de la transformation qui permet de passer d'un triangle au suivant est une rotation d'angle 120 ° dans le sens anti-horaire et de centre le sommet commun à tous les triangles.

11 Wallis et Futuna, décembre 2017, série générale

Références

Série : générale

Académie : Wallis et Futuna

Date : décembre 2017

Exercice 4 - 5 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- rosace
- grandeurs et mesures
- rotation

Thèmes informatiques

- scène

Compétences travaillées

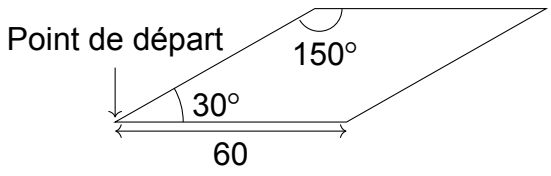
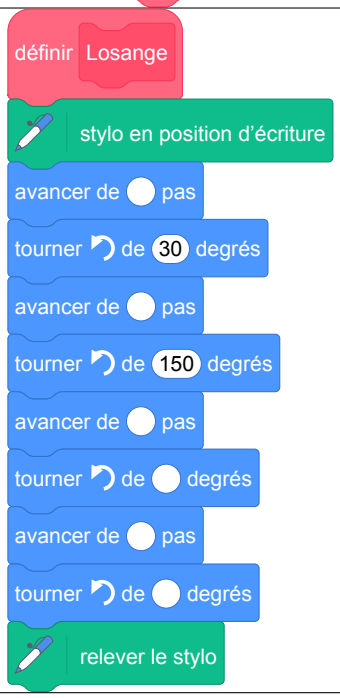
- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme
- écrire un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

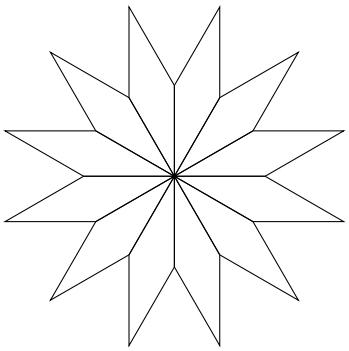
Énoncé - Wallis et Futuma, Décembre 2017

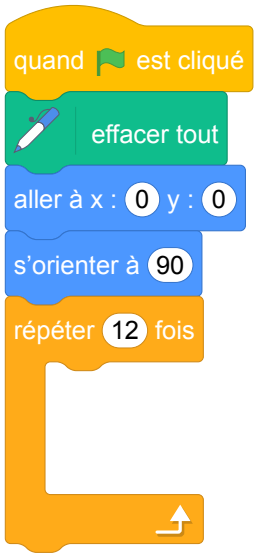
1. On souhaite tracer le motif ci-dessous en forme de losange.

   Compléter ci-dessous, le script du bloc Losange afin d'obtenir ce motif.

Le motif **Losange**	Le bloc Losange
	

2. On souhaite réaliser la figure ci-dessous construite à partir du bloc Losange complété à la question 1.





On rappelle que l'instruction s'orienter à 90 signifie que l'on se dirige vers la droite.

Parmi les instructions ci-dessous, indiquer sur votre copie, dans l'ordre, les deux instructions à placer dans la boucle ci-contre pour finir le script ci-contre.

①	tourner de 30 degrés
③	Losange

②	tourner de 150 degrés
④	avancer de 60 pas

Correction

Fichier scratch : `sg_wallis_futuna_2017_decembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/772316129`

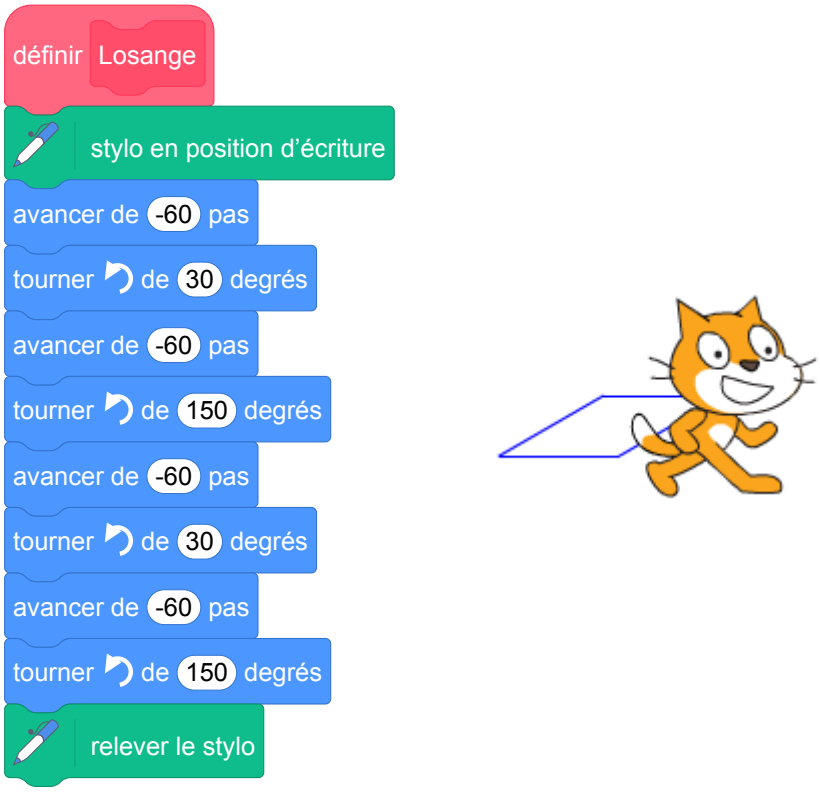
1. Le programme complété :

2. Les deux instructions à placer dans la boucle pour finir le script sont les instructions ③, puis ①.

Analyse

— Cet exercice comporte un travail mathématique intéressant sur les angles supplémentaires. En effet, les réponses attendues sont les valeurs des angles extérieurs à la figure. Ce type d'activité est une occasion de faire vivre ce vocabulaire, mais qui n'est pas exploitée par l'énoncé de brevet.

— Le schéma de losange présent dans l'énoncé ne précise pas l'orientation initiale du lutin. Par conséquent il serait normal d'admettre des réponses peu évidentes a priori, comme le script suivant, qui provoque le tracé situé à droite :



Pour éviter toute ambiguïté sur la réponse attendue, il paraît alors souhaitable que l'orientation initiale du lutin par rapport à la figure à tracer soit précisée.

— La question 2, demandant de compléter le script de production de la rosace, contient lui aussi une ambiguïté puisque le tracé peut se faire indifféremment à l'aide d'une rotation de 30° ou de 150°. Un choix différent des blocs à insérer aurait été souhaitable. Il aurait également été intéressant de faire travailler l'élève sur le nombre de répétitions à utiliser.

Enfin, un travail de justification du choix de l'angle et du nombre de répétitions aurait permis de mettre à l'épreuve les connaissances géométriques et arithmétiques de base et la capacité d'argumentation de l'élève.

12 Métropole – La Réunion , juin 2017, série professionnelle

Références

Série : professionnelle

Académie : Métropole – La Réunion

Date : juin 2017

Exercice 7 - 5 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- repérage
- grandeurs et mesures
- proportionnalité

Thèmes informatiques

- itération bornée

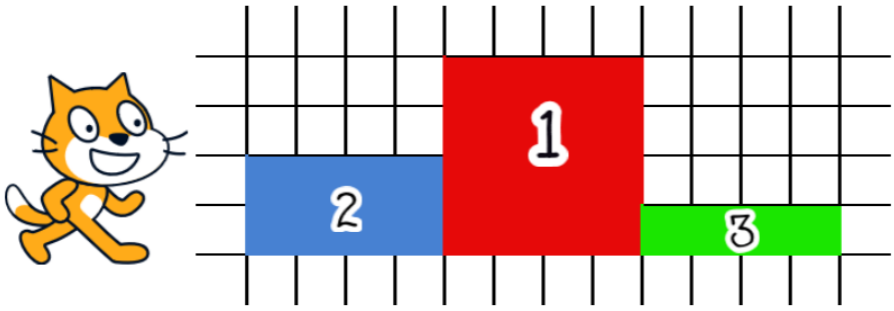
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion, série professionnelle juin 2017

Le chat doit dessiner le podium ci-dessous constitué de trois plots numérotés de 1 à 3 :



1. Les programmes suivants permettent de réaliser les dessins des contours de deux plots qui constituent ce podium. Attribuer à chacun des deux programmes le numéro du plot qui lui correspond. Justifier.

Programme A

```
quand [drapeau] est pressé
s'orienter à 90
stylo en position écriture
aller à x : 80 y : 0
répéter 2 fois
    avancer de 40 pas
    tourner ↺ de 90 degrés
    avancer de 10 pas
    tourner ↺ de 90 degrés
stop tout
```

Programme B

```
quand [drapeau] est pressé
s'orienter à 90
stylo en position écriture
aller à x : 40 y : 0
répéter 4 fois
    avancer de 40 pas
    tourner ↺ de 90 degrés
stop tout
```

Détail de la commande

```
s'orienter à 90
```

(90 ˚) à droite
(−90 ˚) à gauche
(0 ˚) vers le haut
(180 ˚) vers le bas

2. Dans la boucle « répéter » du programme A, quelle instruction faut-il changer pour dessiner le contour du plot manquant ? Indiquer la nouvelle écriture de l'instruction ainsi modifiée.

Correction

Fichier scratch : `pro_metropole_la_reunion_2017_juin.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/362420111/`

1. — Le programme A dessine un rectangle de longueur 40 unités Scratch et le largeur 10 unités Scratch. Le seul plot ayant ces proportions est le plot n°3.
   — Le programme B dessine un carré de côté 40, il s'agit donc du plot n°1.
2. Il faut remplacer

par

pour obtenir le plot n°2.

Analyse

— La situation de départ de ce problème peut paraître déroutante pour les élèves. La figure représentée dans l'énoncé n'est pas celle qu'il faut réellement réaliser. On demande de réaliser le contour de celle-ci.

— Dans l'initialisation de chaque programme, le stylo est mis en position d'écriture avant le retour à la position initiale du lutin. Un tracé parasite peut donc apparaître sur l'écran.

— L'utilisation de l'instruction

après chaque programme est inutile.

— Dans la question 2, il n'y a pas qu'une seule instruction à modifier mais deux : la hauteur du plot bien sûr mais aussi la position initiale qui doit être modifiée.

En conclusion, on peut dire que cet exercice peut manquer d'intérêt d'un point de vue informatique.

Propositions

Nous pouvons renvoyer le lecteur vers le sujet d'Amérique du Sud de novembre 2017 (série générale) qui semble beaucoup plus riche.

13 Polynésie, juin 2017, série professionnelle

Références

Série : générale

Académie : Polynésie

Date : juin 2017

Exercice 6 - nombe de points inconnus

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- agrandissement réduction
- caractérisation des figures
- frise
- grandeurs et mesures

Thèmes informatiques

- variable déclarée
- activité du lutin
- calcul
- itération conditionnelle
- temps

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	1
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	1
Etat	0

Énoncé - Polynésie, Juin 2017, série professionnelle

Lors du triathlon international de Papeete, une entreprise de sécurité, Sécudrone, souhaite effectuer la surveillance vidéo du parc à vélo à l'aide d'un drone équipé d'une caméra. Sécudrone veut programmer les déplacements du drone. L'algorithme du déplacement du drone est testé à l'aide du logiciel scratch. Le programme est donné ci-dessous :

```
quand [drapeau] est cliqué
s'orienter à 90
aller à x : -200 y : -200
effacer tout
mettre L à 0
mettre l à 0
stylo en position d'écriture
répéter jusqu'à ce que (L = 400) ou (l = 200)
    ajouter 40 à L
    ajouter 20 à l
    avancer de L pas
    tourner de ↺ de 90 degrés
    attendre 1 secondes
    avancer de l pas
    tourner de ↺ de 90 degrés
    attendre 1 secondes
    avancer de L pas
    tourner de ↺ de 90 degrés
    attendre 1 secondes
    avancer de l pas
    tourner de ↺ de 90 degrés
    attendre 1 secondes
```

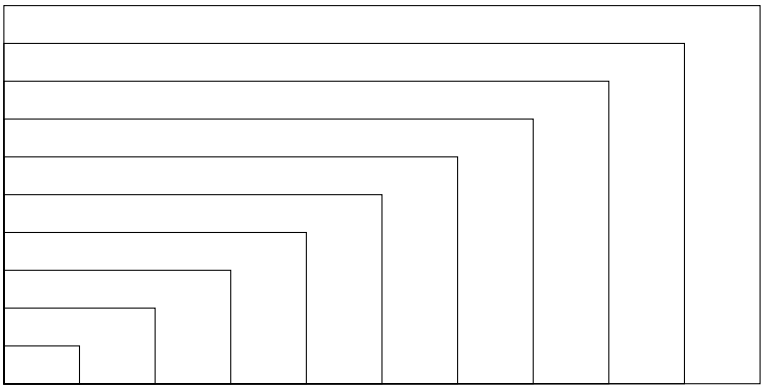
1. Le drone va effectuer plusieurs boucles rectangulaires pendant sa surveillance vidéo. Dessiner à main levée le motif obtenu.
2. Combien de boucles rectangulaires va-t-il décrire selon cet algorithme ?
3. Que fera le drone, si on modifie dans l'algorithme ces deux lignes :

```
ajouter 80 à L
ajouter 40 à l
```

Correction

Fichier Scratch : `pro_polynesie_2017_juin.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/480986861`

1. 

2. Il y aura 10 boucles rectangulaires. En effet, la condition d'arrêt est $L = 400$ ou $l = 200$. Ces deux variables sont initialisées à zéro. À chaque itération, L est augmentée de 40 et l est augmentée de 20.

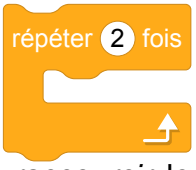
$$400 \div 40 = 200 \div 20 = 10$$

3. Les cinq rectangles de la figure obtenue par le deuxième programme sont deux fois plus grands que leurs homologues obtenus par le premier programme. Comme la condition d'arrêt n'a pas changé, il y a deux fois moins de rectangles (5 au lieu de 10). Il ne reste que les rectangles de rang pair par rapport au programme précédent.

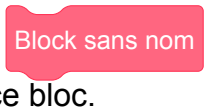
Analyse

— Même si dans ce programme, la condition d'arrêt fonctionne parce que les variables sont de type entier et que leurs modifications successives leur permet d'atteindre cette condition, elle est mal choisie d'un point de vue général. En effet, l'usage d'une inégalité large (supérieure ou égale) au lieu d'une égalité stricte assurera l'arrêt de la boucle dans tous les cas et en particulier en cas d'incrémentation des valeurs des variables flottantes donc l'écriture binaire peut être illimitée et donc conduire à des approximations dans la machine.

— Il aurait été préférable de donner des noms vraiment explicites aux variables, par exemple longueur et largeur au lieu de L et l, avec le l qui en plus ressemble à un i majuscule.

— Il est dommage que les instructions du tracé du rectangle n'utilisent pas une boucle 

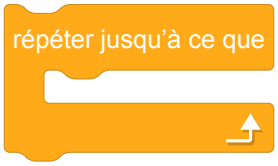
ou qu'il n'y ait pas à une question sur ce point en demandant à l'élève comment *raccourcir* le programme par exemple.

— Ce sujet pourrait également utiliser un bloc 

pour le dessin du rectangle ce qui permettrait de demander à l'élève le rôle de ce bloc.

Ces deux dernières propositions peuvent être posées avant la question 1 et permettrait d'aider les élèves à y répondre.

— Il est possible de prolonger l'activité avec un travail autour de l'homothétie (le 3e programme dans le studio Scratch). Il est intéressant de faire distinguer aux élèves l'évolution arthmétique d'une évolution géométrique des variables longueur et largeur en leur proposant de modifier le programme initial pour obtenir une figure avec des rectangles homothétiques. Le programme, une fois rédigé serait à tester sur machine.

Si on garde dans la première partie l'égalité comme condition d'arrêt, il est possible, dans la partie sur l'homothétie, de laisser cette condition en blanc et de la faire compléter par les élèves. Les élèves seront alors tentés de remettre le « égal » mais l'activité sur machine montrera que cela ne fonctionne pas en gardant les mêmes bornes avec une égalité. On les conduira ainsi à choisir le supérieur ou égal dans la condition d'arrêt. On pourra également les questionner sur le choix du type de boucle : 

ou

.

Par ailleurs, dans le cadre des rectangles homothétiques, seule la donnée de la longueur (ou de la largeur) initiale est absolument nécesaire. Il est possible d'enrichir l'exercice mathématiquement en faisant mesurer aux élèves les dimensions de la figure pour leur faire déterminer le ratio largeur/longueur afin de calculer la valeur d'une de ces variables à partir de l'autre.

Propositions

Énoncé modifié

Lors du triathlon international de Papeete, une entreprise de sécurité, Sécudrone, souhaite effectuer la surveillance vidéo du parc à vélo à l'aide d'un drone équipé d'une caméra. Sécudrone veut programmer les déplacements du drone. L'algorithme du déplacement du drone est testé à l'aide du logiciel scratch. Le programme est donné ci-dessous :

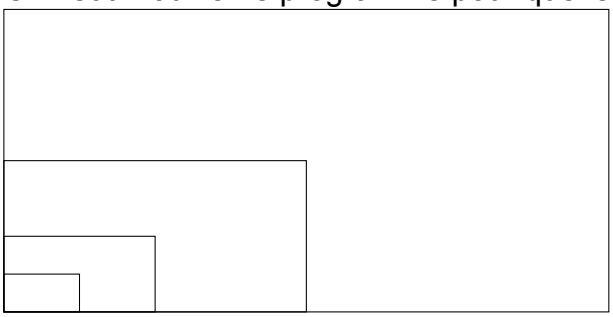
```
quand [drapeau] est cliqué
s'orienter à 90
aller à x : -200 y : -200
effacer tout
mettre longueur à 0
mettre largeur à 0
stylo en position d'écriture
répéter jusqu'à ce que (longueur = 400) ou (largeur = 200)
    ajouter 40 à longueur
    ajouter 20 à largeur
    Bloc sans nom (longueur) (largeur)
```

```
définir Bloc sans nom (x) (y)
avancer de (x) pas
tourner de ↺ de (90) degrés
attendre (1) secondes
avancer de (y) pas
tourner de ↺ de (90) degrés
attendre (1) secondes
avancer de (x) pas
tourner de ↺ de (90) degrés
attendre (1) secondes
avancer de (y) pas
tourner de ↺ de (90) degrés
attendre (1) secondes
```

1. Le drone va effectuer plusieurs circuits similaires, chacun correspondant à l'exécution du `Bloc sans nom (x) (y)` pendant sa surveillance vidéo. Dessiner à main levée le motif obtenu lors de l'exécution de ce bloc. Quel nom plus explicite pourriez-vous donner à ce bloc ?
2. Réécrire le `Bloc sans nom (x) (y)` pour qu'il permette de réaliser le même déplacement mais avec un script plus court.
3. Combien de motifs `Bloc sans nom (x) (y)` va-t-il décrire selon cet algorithme ?
4. Dessiner à main levée le motif obtenu lors du trajet complet du drone.
5. Que fera le drone, si on modifie dans l'algorithme ces deux lignes :

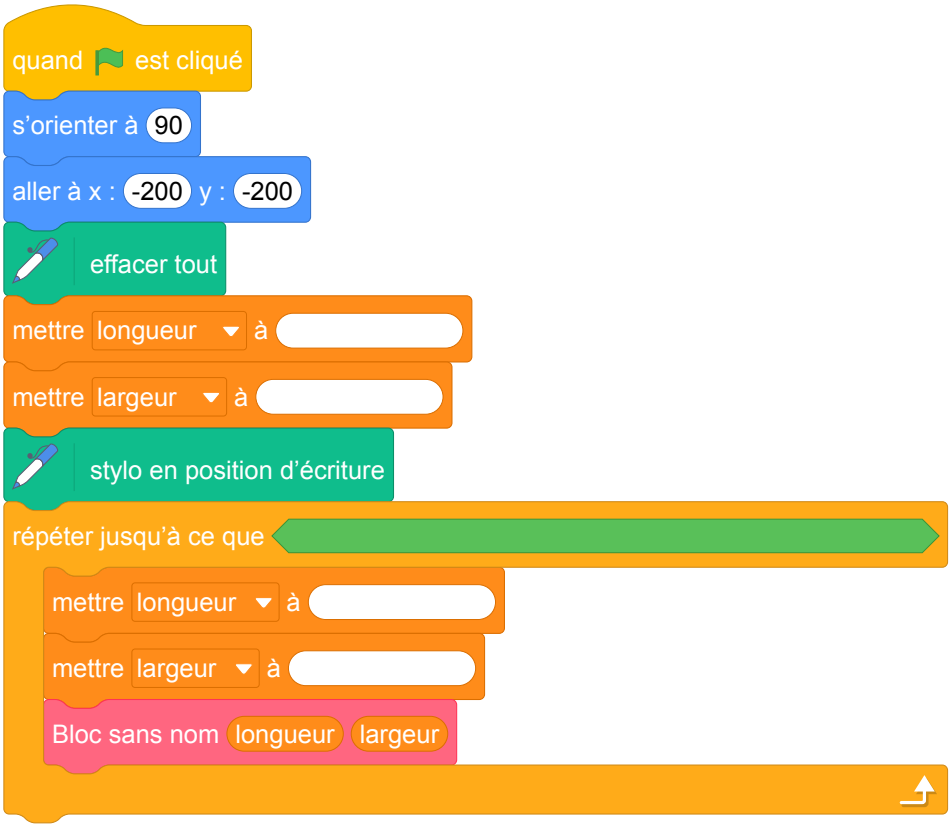
```
ajouter 80 à longueur
ajouter 40 à largeur
```

6. On veut modifier le programme pour que le drone parcoure la trajectoire suivante :



On suppose que le drone part du coin en bas à gauche de la figure et qu'elle commence par le petit rectangle. Quelle transformation géométrique permet de passer d'un rectangle au rectangle suivant ?

7. Compléter le script suivant pour que le drone parcoure cette trajectoire.



Correction de l'énoncé modifié

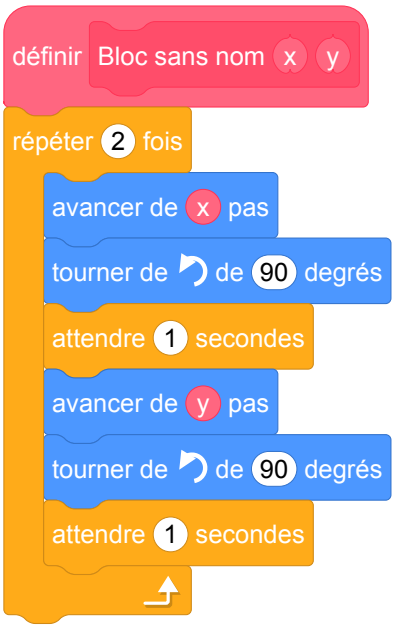
Fichier Scratch : `pro_polynesie_2017_juin_modifie.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/507996687`

1.

   Le bloc permet de dessiner un rectangle de côtés x et y.

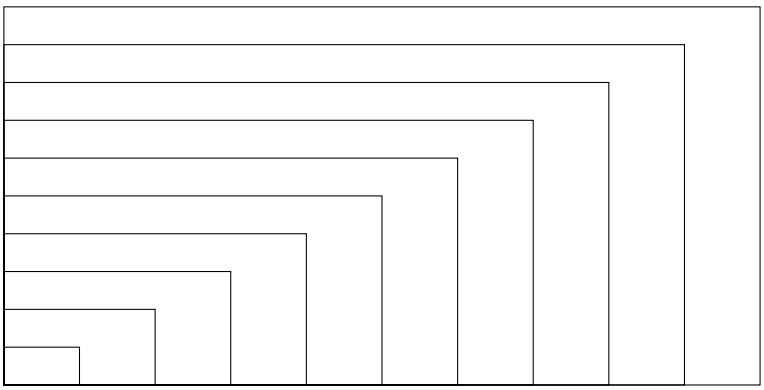
   Il peut être renommé Dessiner un rectangle x y.

2. On remarque que dans ce bloc apparaît 2 fois la même série d'instructions. On peut donc utiliser une structure de contrôle répéter 2 fois.



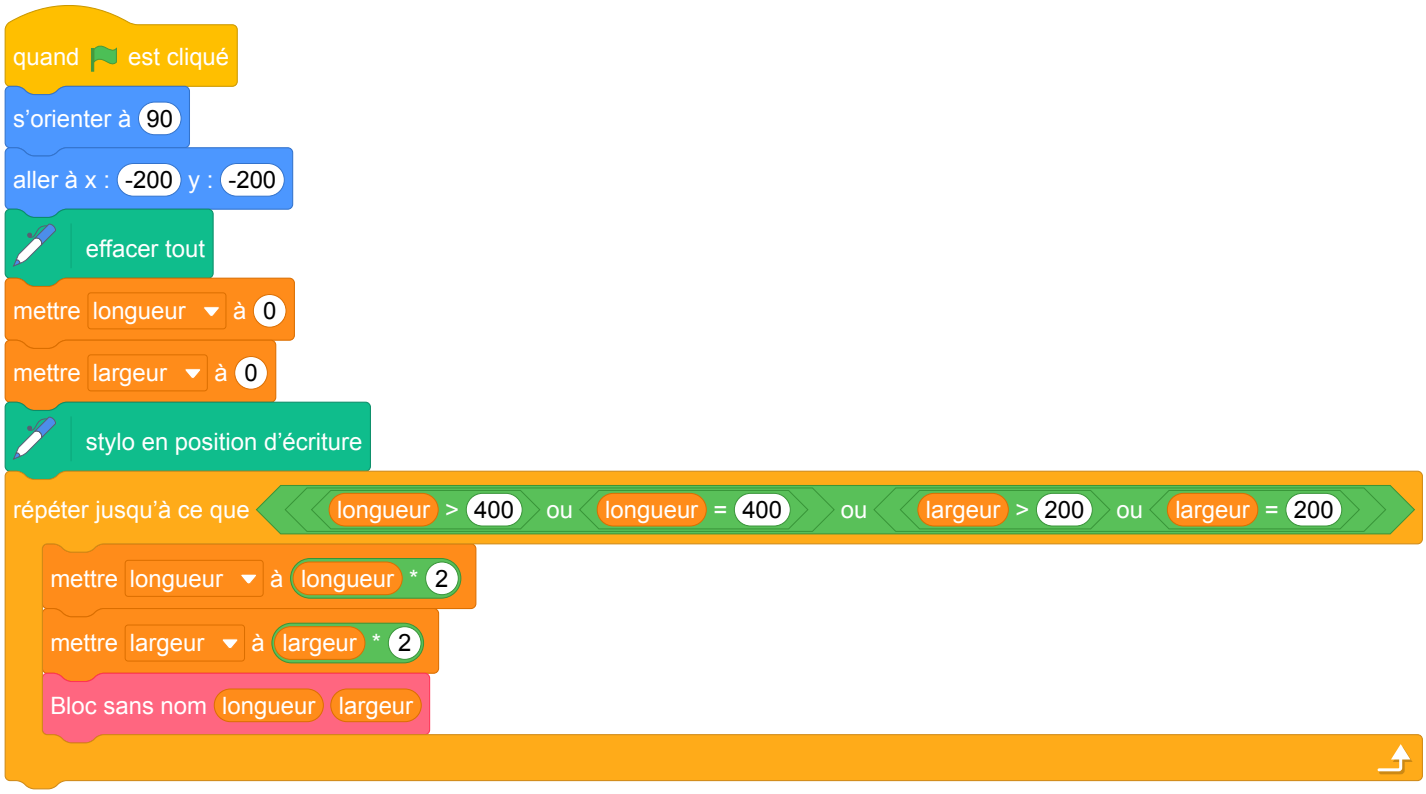
3. Il y aura 10 boucles rectangulaires. En effet, la condition d'arrêt est $L = 400$ ou $l = 200$. Ces deux variables sont initialisées à zéro. À chaque itération, L est augmentée de 40 et l est augmentée de 20.

$$400 \div 40 = 200 \div 20 = 10$$

4. 

5. Les cinq rectangles de la figure obtenue par le deuxième programme sont deux fois plus grands que leurs homologues obtenus par le premier programme. Comme la condition d'arrêt n'a pas changé, il y a deux fois moins de rectangles (5 au lieu de 10). Il ne reste que les rectangles de rang pair par rapport au programme précédent.

6. Il s’agit d’une homothétie ayant pour centre le point de départ du drone et pour rapport 2.
7.



14 Métropole – La Réunion – Antilles – Guyane, septembre 2017, série professionnelle

Références

Série : professionnelle

Académie :Métropole – La Réunion – Antilles – Guyane

Date : septembre 2017

Exercice 6 - 5 points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- repérage
- déplacement
- translation

Thèmes informatiques

- variable réservée
- activité du lutin
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion – Antilles – Guyane, septembre 2017

1. Associer chaque programme A, B, C à la figure 1 ou 2 qu'il permet de réaliser.

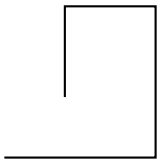


Figure 1

Figure 2

Remarque : les figures ne sont pas à l'échelle

Détail de la commande

s'orienter à 90

(90 ˚) à droite
(–90 ˚) à gauche
(0 ˚) vers le haut
(180 ˚) vers le bas

Programme A

```
quand [drapeau] est cliqué
stylo en position d'écriture
s'orienter à 90
avancer de 50 pas
tourner ↺ de 90 degrés
avancer de 50 pas
tourner ↺ de 90 degrés
avancer de 30 pas
tourner ↺ de 90 degrés
avancer de 30 pas
stop tout
```

Programme B

```
quand [drapeau] est cliqué
stylo en position d'écriture
s'orienter à 90
répéter 2 fois
    avancer de 50 pas
    tourner ↺ de 90 degrés
    avancer de 30 pas
    tourner ↺ de 90 degrés
stop tout
```

Programme C

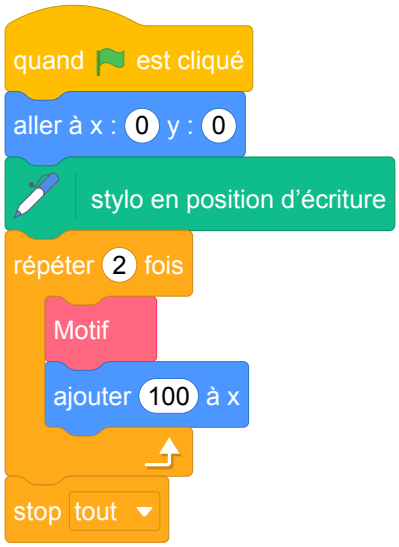
```
quand [drapeau] est cliqué
stylo en position d'écriture
s'orienter à 90
répéter 2 fois
    avancer de 50 pas
    tourner ↺ de 90 degrés
répéter 2 fois
    avancer de 30 pas
    tourner ↺ de 90 degrés
stop tout
```

2. Quelle est la nature du quadrilatère dessiné par le bloc Motif ?

```
définir Motif
stylo en position d'écriture
répéter 4 fois
    avancer de 70 pas
    tourner ↺ de 90 degrés
```

3. Le programme D ci-dessous permet-il d'obtenir la figure 3 ou la figure 4 ? Pourquoi ?
Remarque : les figures ne sont pas à l'échelle.

Programme D



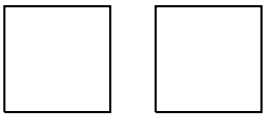


Figure 3

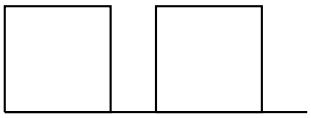


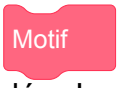
Figure 4

Correction

Fichier scratch : `pro_metropole_2017_septembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/362421616`

1. La figure 1 peut être obtenue par les programmes A et C.
   La figure 2 peut être obtenue par le programme B.
2. Le bloc motif permet de dessiner un carré.
3. Étant donné que le stylo reste en position d'écriture tout le long du sujet, le programme C permet d'obtenir la figure 4.

Analyse

— C'est l'exécution du bloc 

qui produit le carré et non pas le bloc lui même. La question 2 semble donc mal formulée. La formulation "Quelle est la nature du quadrilatère dessiné par l'exécution du bloc

?" serait préférable.

— L'utilisation de l'instruction

est inutile.

— Dans le programme D, il n'y a plus d'instruction sur l'orientation initiale du lutin. Cela semble important pour un sujet de DNB, mais pour une utilisation en classe cela pourrait permettre de discuter avec les élèves de l'état initial d'un programme.

En conclusion, nous pouvons dire que cet exercice peut manquer d'intérêt informatique si ce n'est dans l'utilisation d'un bloc d'instructions.

Propositions

On pourrait imaginer dans un travail en classe, et non dans un sujet de DNB, de modifier la position des carrés de manière penchée, dans la question 3, et ce afin de travailler sur une position non prototypique des carrés.

15 Polynésie, septembre 2017, série professionnelle

Références

Série : professionnelle

Académie : Polynésie

Date : septembre 2017

Exercice 5 - ? points sur 50

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie sans transformation
- déplacement

Thèmes informatiques

- variable déclaree
- communication
- activité du lutin
- branchement conditionnel
- itération bornée
- temps
- capteurs

Compétences travaillées

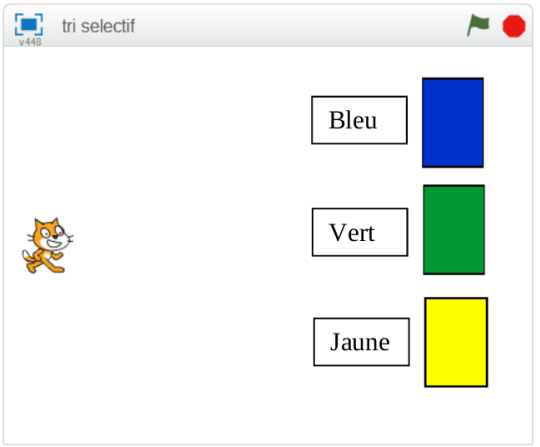
- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	1
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

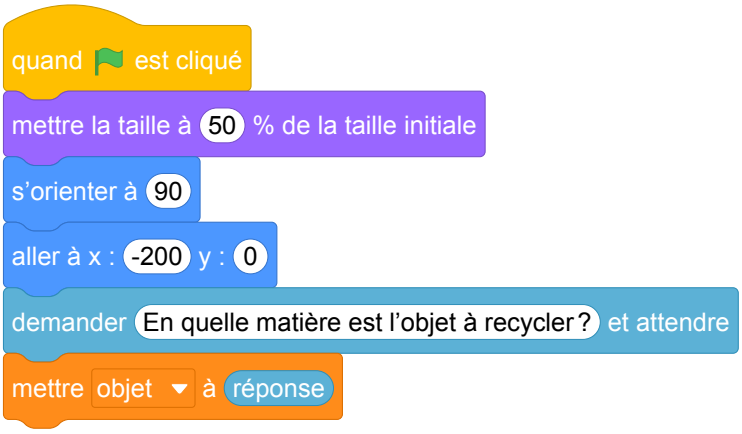
Énoncé - Polynésie, Septembre 2017

Avec le logiciel Scratch, on a créé un programme permettant au chat de déposer dans la bonne poubelle les objets à recycler :

- le carton dans la poubelle bleue,
- le verre dans la poubelle verte,
- le plastique dans la poubelle jaune.



Début du programme

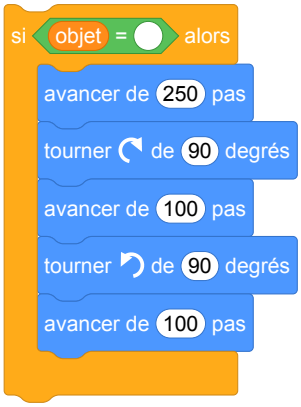


1. Donner les coordonnées du chat au départ.
2. La suite du programme est composée de trois groupes d'instructions : A, B et C.
   Dans chaque groupe, il manque un mot.
   Attribuer à chaque groupe d'instructions le mot manquant parmi les suivants :

carton / verre / plastique.

A

B



C

3. Dans le programme ci-dessous, l'instruction

a été ajoutée pour que le chat puisse jeter 5 objets.
Cette instruction a-t-elle été placée au bon endroit ? Justifier la réponse.

```
quand [drapeau] est cliqué
mettre la taille à 50 % de la taille initiale
s'orienter à 90
aller à x : -200 y : 0
demander En quelle matière est l'objet à recycler ? et attendre
répéter 5 fois
    mettre objet à réponse
    si <objet = ●> alors
        avancer de 350 pas
    si <objet = ●> alors
        avancer de 250 pas
        tourner ↻ de 90 degrés
        avancer de 100 pas
        tourner ↺ de 90 degrés
        avancer de 100 pas
    si <objet = ●> alors
        avancer de 250 pas
        tourner ↺ de 90 degrés
        avancer de 100 pas
        tourner ↻ de 90 degrés
        avancer de 100 pas
    attendre 1 secondes
```

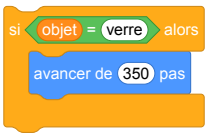
Correction

Fichier scratch : `pro_polynesie_2017_septembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/362425848/`

1. Les coordonnées du chat au départ sont $(-200; 0)$.
2. Voici l'attribution

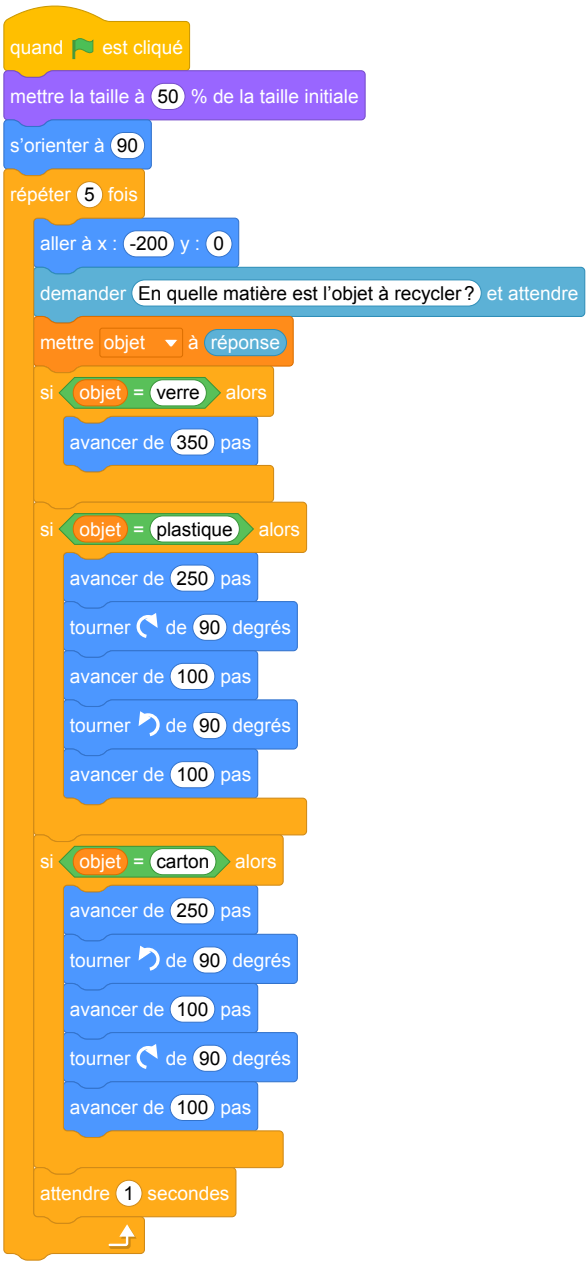
A



B

C

3. L'instruction n'a pas été placée au bon endroit, il est nécessaire de reposer la question à chaque répétition et de replacer le lutin à sa position de départ.



Analyse

— Dans la question 1, on demande les coordonnées du chat au départ, or nous ne les connaissons pas. Par contre, nous les connaissons après l'exécution du début du programme.

— Dans la question 2, le fait de devoir répondre "carton" peut être une difficulté sémantique pour les élèves : est-ce un carton, en carton, du carton, etc. ? Le nom de la variable n'est peut être pas non plus bien choisi. On aurait pu choisir "matièreObjet" par exemple. Par contre, on souligne le fait de nommer une variable avec un nom explicite.

— L'utilisation très locale de la variable réponse est une bonne pratique.

— Dans la question 3, on peut soulever la question de l'utilisation d'un enchaînement de "si ... alors" plutôt que des "si ... alors ... sinon". On remarque que cela permet, dans cet exercice, la réutilisation des morceaux d'instructions de la question précédente.

Propositions

Étant donné la remarque précédente, il serait intéressant de prolonger le travail par l'utilisation du bloc "si ... alors ... sinon".

Brevets 2018

1 Polynésie, juillet 2018, série générale

Références

Série : générale

Académie : Polynésie

Date : juillet 2018

Exercice 6 - 14 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable déclarée
- calcul
- évènement avec message
- capteurs

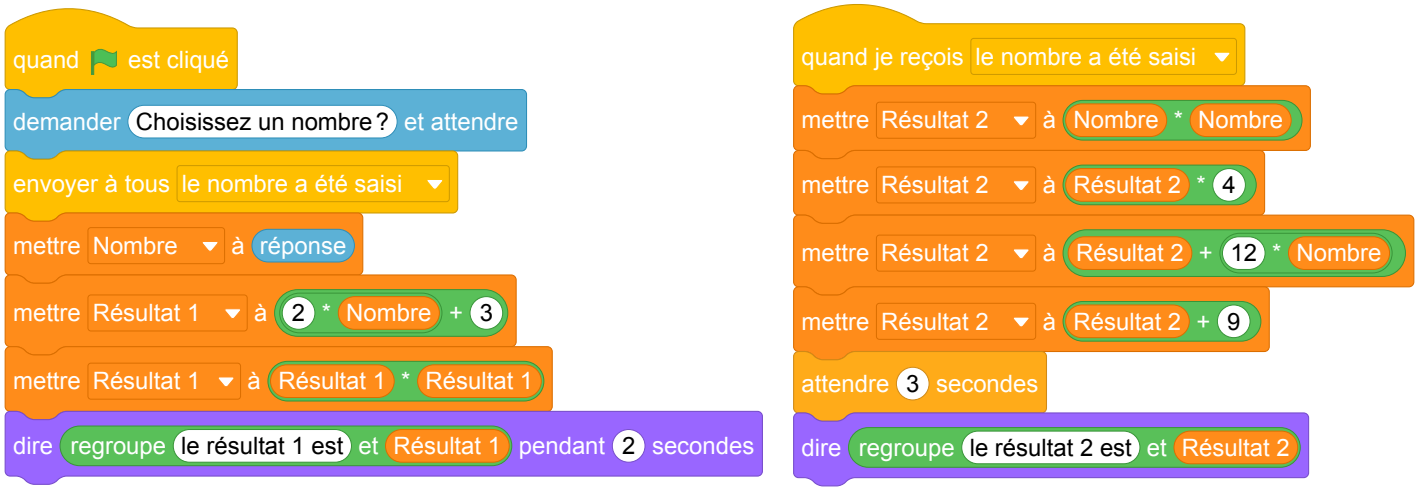
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	1
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, juillet 2018

Voici un script saisi par Alice dans un logiciel d'algorithmique.



1. Alice a choisi 3 comme nombre, calculer les valeurs de Résultat 1 et de Résultat 2 ?
   Justifier en faisant apparaître les calculs réalisés.
2. Généralisation
   (a) En appelant x le nombre choisi dans l'algorithme, donner une expression littérale traduisant la première partie de l'algorithme correspondant à Résultat 1.
   (b) En appelant x le nombre choisi dans l'algorithme, donner une expression littérale traduisant la deuxième partie de l'algorithme correspondant à Résultat 2.
   (c) Trouver le ou les nombres choisis par Alice qui correspondent au résultat affiché ci-dessous.

Correction

Fichier Scratch : `sg_polynesie_2018_juillet.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/386282655/`

1. Le Résultat 1 prend d'abord la valeur $2 * 3 + 3 = 9$ ensuite il prend la valeur $9 \times 9 = 81$.
   Le Résultat 2 prend d'abord la valeur $3 \times 3 = 9$, puis $9 \times 4 = 36$, puis $36 + 12 \times 3 = 72$ et enfin $72 + 9 = 81$.
2. (a) $(2x + 3)^2$
   (b) $4x^2 + 12x + 9$
   (c) D'après le deuxième algorithme, on a $4x^2 + 12x + 9 = 9$ donc $4x^2 + 12x = 0$. Or on peut factoriser cette expression $4x(x + 3) = 0$ d'où $4x = 0$ ou $x + 3 = 0$. On trouve finalement $x = 0$ ou $x = -3$. Alice a choisi soit 0 soit -3.

Analyse

L'exercice propose une programmation parallèle pour deux programmes de calcul comme vus au collège en mathématiques pour aborder le calcul littéral. Les variables Résultat 1 et Résultat 2 sont calculées en même temps après que le message « le nombre a été saisi »est diffusé.

Par contre, dans le programme original il faut échanger l'envoi du message et « mettre Nombre à ... ». En effet, dans la séquence d'instructions donnée par l'énoncé, le calcul du « Résultat 2 »peut commencer avant que la variable « Nombre »soit mise à jour et donc avec une valeur préexistante.

Pour homogénéiser les deux programmes Scratch et la question 2 de l'exercice, il serait plus judicieux d'utiliser une variable nommée x plutôt que Nombre.

Le programme de l'énoncé peut donc être remplacé par :

```
quand [drapeau] est cliqué
demander [Choisissez un nombre ?] et attendre
mettre [x] à [réponse]
envoyer à tous [le nombre x a été saisi]
mettre [Résultat 1] à [2 * x + 3]
mettre [Résultat 1] à [Résultat 1 * Résultat 1]
dire [regroupe [le résultat 1 est] et [Résultat 1]] pendant [2] secondes
```

```
quand je reçois [le nombre x a été saisi]
mettre [Résultat 2] à [x * x]
mettre [Résultat 2] à [Résultat 2 * 4]
mettre [Résultat 2] à [Résultat 2 + 12 * x]
mettre [Résultat 2] à [Résultat 2 + 9]
attendre [3] secondes
dire [regroupe [le résultat 2 est] et [Résultat 2]]
```

Il faudra également expliquer aux élèves le but d'attendre 3 secondes dans le second programme de calcul afin de pouvoir afficher le texte souhaité sur la scène. Le point positif est qu'on s'intéresse à la communication entre processus pour laquelle il faudrait trouver une autre situation d'accroche.

Dans la question 1, on attend que l'élève sache le fonctionnement de Scratch avec la programmation parallèle (deux programmes de calcul exécutés simultanément) et la programmation événementielle (envoi des messages, instructions déclencheurs). Cette connaissance lui permet de comprendre que le programme Scratch exécute en vrai deux programmes de calcul. Nous pouvons nous poser la question de l'utilité de ce type de programmation dans le cadre de cette question. Les connaissances mathématiques qui sont évaluées ne dépendent pas des types de programmation proposés. Les mêmes calculs peuvent se faire en séquentiel (voir la section Propositions). Si l'utilisation de la programmation parallèle doit être maintenue on pourrait aller au bout de cette idée avec un retour de message à la fin des calculs par exemple ou bien attribuer les deux calculs aux deux lutins différents.

Dans la question 2, les appellations "première partie" et "deuxième partie" ne sont pas pertinentes. On pourrait simplement parler du calcul du Résultat 1 et du Résultat 2.

D'autre part, cet exercice met l'accent sur la notion d'affectation en informatique et sur les égalités littérales en mathématiques. La différence entre ces deux notions est à travailler avec les élèves pour que ces derniers puissent faire la transposition d'une notion sur l'autre. Il est important de remarquer que l'élève qui n'a pas compris le programme Scratch ne pourra pas forcément faire le calcul littéral même s'il est à l'aise dans ce domaine, d'autant plus que dans ce sujet de brevet il n'y a pas d'autre exercice de calcul littéral.

Propositions

Pour que les deux programmes de calcul puissent commencer en même temps, il aurait fallu inverser les deux instructions suivantes :

envoyer à tous le nombre a été saisi

mettre Nombre à réponse

En effet, dans le programme proposé, il est théoriquement possible que le calcul du Résultat 2 commence alors que la variable Nombre n'a pas été initialisée. Même si en pratique le programme marche, il est important de suivre un ordre cohérent dans les instructions.

Pour homogénéiser les deux programmes Scratch et la question 2 de l'exercice, il serait plus judicieux d'utiliser une variable nommée x plutôt que Nombre. On peut également unifier les variables Résultat 1 et Résultat 2 et n'utiliser qu'une seule variable pour les calculs. En effet, s'il y avait, par exemple, une comparaison des deux résultats, il serait important de les nommer différemment.

Le même exercice peut très bien être proposé avec un programme séquentiel avec un programme qui suit :

```
quand [drapeau] est cliqué
demander Choisissez un nombre ? et attendre
mettre x à réponse
mettre Résultat à (2 * x) + 3
mettre Résultat à Résultat * Résultat
dire (regroupe le résultat 1 est et Résultat) pendant 2 secondes
mettre Résultat à x * x
mettre Résultat à Résultat * 4
mettre Résultat à Résultat + (12 * x)
mettre Résultat à Résultat + 9
dire (regroupe le résultat 2 est et Résultat)
```

2 Polynésie, septembre 2018, série générale

Références

Série : générale

Académie : Polynésie

Date : septembre 2018

Exercice 6 - 12 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- calcul sans programme
- proportionnalité

Thèmes informatiques

- variable déclarée
- calcul
- itération conditionnelle

Compétences travaillées

- comprendre un algorithme

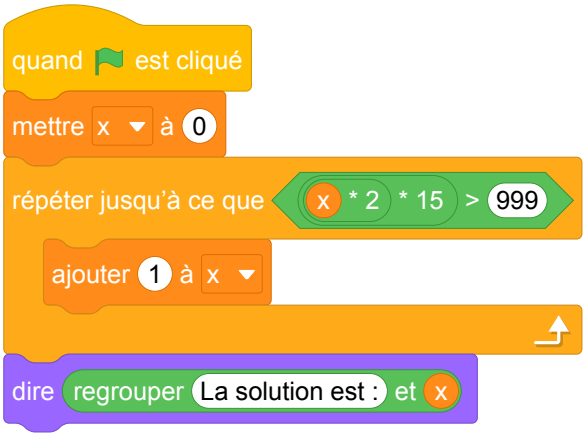
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	1
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	1
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, septembre 2018

Une personne pratique le vélo de piscine depuis plusieurs années dans un centre aquatique à raison de deux séances par semaine. Possédant une piscine depuis peu, elle envisage d'acheter un vélo de piscine pour pouvoir l'utiliser exclusivement chez elle et ainsi ne plus se rendre au centre aquatique.

- Prix de la séance au centre aquatique : 15 €.
- Prix d'achat d'un vélo de piscine pour une pratique à la maison : 999 €.

1. Montrer que 10 semaines de séances au centre aquatique lui coûtent 300 €.
2. Que représente la solution affichée par le programme ci-après ?



3. Combien de semaines faudrait-il pour que l'achat du vélo de piscine soit rentabilisé ?

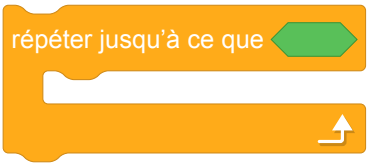
Correction

Fichier Scratch : `sg_polynesie_2018_septembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/772427559/`

1. 10 semaines de séances au centre aquatique lui coûtent : $2 \times 10 \times 15 = 30 \times 10 = 300$ €.
2. La solution donne le nombre minimal de semaines au bout duquel le coût des séances au centre aquatique dépassera le coût d'achat du vélo.
3. Il faut donc résoudre l'inéquation : $2 \times 15 \times x > 999$ ou
   $30x > 999$ soit $x > \dfrac{999}{30}$ ou $x > \dfrac{333}{10}$ soir $x > 33,3$.
   L'achat du vélo sera rentabilisé à partir de la 34e semaine.

Analyse

L'exercice consiste à résoudre une inéquation avec un programme. La variable x représente un nombre entier. L'exercice demande de comprendre le programme et donc de réfléchir à son sens et ensuite résoudre l'inéquation proposée pour trouver le résultat qui sera affiché par le programme. Il est également à noter qu'il est assez rare de trouver une boucle non bornée



dans les énoncés de brevet.

Propositions

Il est possible de revoir cet exercice pour travailler avec des fonctions affines et linéaires, en utilisant un nouveau bloc avec paramètres pour chacune des fonctions. En 2nde, il est aussi possible d'étendre et d'adapter ces types d'exercice pour travailler avec les suites géométriques.

Énoncé : Utilisation des fonctions linéaires / affines

Alain décide d'aller régulièrement au cinéma pendant un mois. Dans son cinéma préféré, il font deux types de promotions :

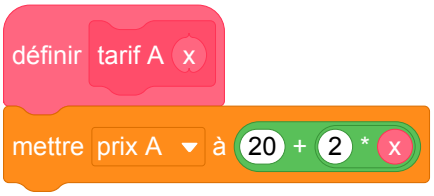
- tarif 1 : 5€ par entrée ;
- tarif 2 : 20€ d'adhésion mensuelle puis 2€ par entrée.

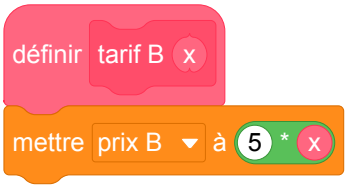
Partie I

1. Quel prix paiera-t-il avec chaque tarif, s'il va au cinéma quatre fois par mois ? Quel tarif sera intéressant dans ce cas ?
2. On appelle x le nombre de fois où Alain ira à la cinéma.
   (a) On note $tarif_1(x)$ le prix qu'il paiera avec le tarif 1. Déterminer $tarif_1(x)$ en fonction de x.
   (b) On note $tarif_2(x)$ le prix qu'il paiera avec le tarif 2. Déterminer $tarif_2(x)$ en fonction de x.

Partie II

1. Parmi les blocs suivants, lequel correspond au tarif 2 ?





2. Voici le programme principal qui utilise les deux blocs précédents.

   Que représente la solution affichée par ce programme ?

3. Le cinéma fait une nouvelle promotion et propose une troisième tarif avec 15€ d'adhésion mensuelle puis 3€ par entrée. Écrire le bloc scratch correspondant au calcul du prix prix 3 à payer avec ce tarif 3.

Correction : Utilisation des fonctions linéaires / affines

Fichier Scratch : `sg_polynesie_2018_septembre_v2.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/772425863/`

Partie I

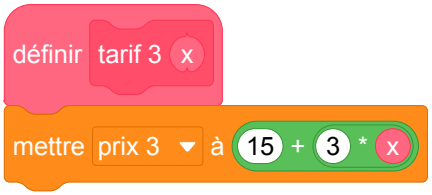
1. Avec le tarif 1, il paiera $5 \times 4 = 20$ €. Avec le tarif 2, il paiera $20 + 2 \times 4 = 28$€. Le tarif 1 est plus intéressant.
2. (a) $tarif_1(x) = 5x$
   (b) $tarif_2(x) = 20 + 2x$

Partie II

1. Le bloc

correspond au tarif 2. Le bloc

correspond au tarif 1.
2. La solution représente le nombre de séances à partir duquel le tarif 2 est plus avantageux que le tarif 1.
3. Voici le bloc scratch demandé.



3 Pondichéry, mai 2018, série générale

Références

Série : générale

Académie : Pondichéry

Date : mai 2018

Exercice 5 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie sans transformation
- repérage
- nombres et calculs
- calcul sans programme
- grandeurs et mesures
- probabilités
- simulations

Thèmes informatiques

- variable réservée
- variable déclarée
- calcul
- aleatoire
- branchement conditionnel
- itération bornée

Compétences travaillées

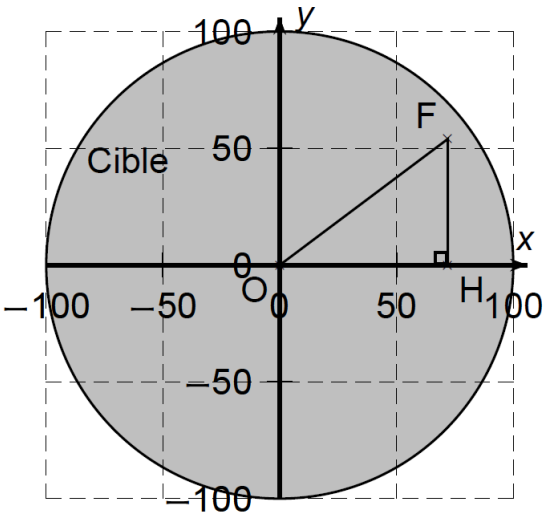
- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	1
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	1
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Pondichéry, 2 mai 2018

Dans tout l'exercice l'unité de longueur est le mm.

On lance une fléchette sur une plaque carrée sur laquelle figure une cible circulaire (en gris sur la figure). Si la pointe de la fléchette est sur le bord de la cible, on considère que la cible n'est pas atteinte.
On considère que cette expérience est aléatoire et l'on s'intéresse à la probabilité que la fléchette atteigne la cible.

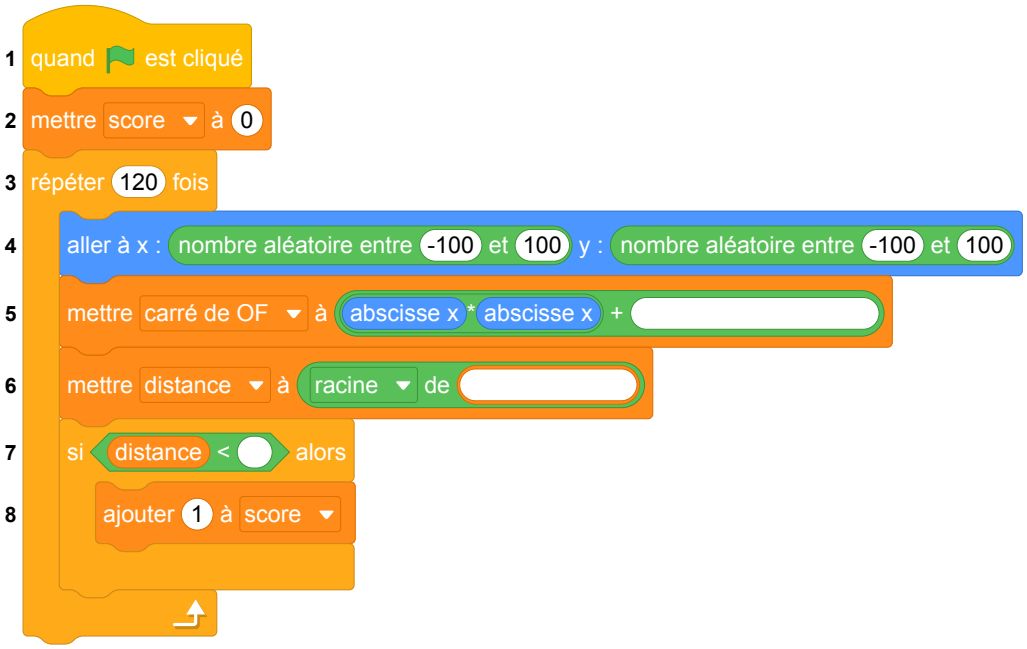


— La longueur du côté de la plaque carrée est 200.

— Le rayon de la cible est 100.

— La fléchette est représentée par le point F de coordonnées $(x\ ;\ y)$ où x et y sont des nombres aléatoires compris entre -100 et 100.

1. Dans l'exemple ci-dessus, la fléchette F est située au point de coordonnées $(72\ ;\ 54)$.
   Montrer que la distance OF, entre la fléchette et l'origine du repère est 90.
2. D'une façon générale, quel nombre ne doit pas dépasser la distance OF pour que la fléchette atteigne la cible ?
3. On réalise un programme qui simule plusieurs fois le lancer de cette fléchette sur la plaque carrée et qui compte le nombre de lancers atteignant la cible. Le programmeur a créé trois variables nommées : carré de OF, distance et score.



   (a) Lorsqu'on exécute ce programme, combien de lancers sont simulés ?
   (b) Quel est le rôle de la variable score ?
   (c) Compléter et recopier sur la copie uniquement les lignes 5, 6 et 7 du programme afin qu'il fonctionne correctement.
   (d) Après une exécution du programme, la variable score est égale à 102.
   À quelle fréquence la cible a-t-elle été atteinte dans cette simulation ?
   Exprimer le résultat sous la forme d'une fraction irréductible.

4. On admet que la probabilité d'atteindre la cible est égale au quotient : aire de la cible divisée par aire de la plaque carrée.
   Donner une valeur approchée de cette probabilité au centième près.

Correction

Fichier scratch : `sg_pondichery_2018_mai.sb3` Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/426264230`

1. Le triangle OFH est rectangle en H ; du théorème de Pythagore appliqué à ce triangle on déduit : $OF^2 = OH^2 + HF^2$, soit $OF^2 = 72^2 + 54^2 = 5\,184 + 2\,916 = 8\,100$, donc $OF = \sqrt{8\,100} = 90$.
2. La fléchette doit être à l'intérieur du cercle, donc on doit avoir $OF < 100$ puisque 100 est le rayon du disque.
   Autrement dit $OF < 100$.
3. (a) On simule 120 lancers.
   (b) score comptabilise le nombre de lancers ayant atteint la cible.
   (c) Dans la ligne 5 il faut compléter par ordonnée y * ordonnée y.
   Dans la ligne 6 il faut écrire carré de OF.
   Dans la ligne 7 il faut compléter avec le nombre 100.
   (d) Le nombre de réussites étant égal à 102 sur 120 lancers, la fréquence de réussite est égale à
   $\frac{102}{120} = \frac{51}{60} = \frac{3 \times 17}{3 \times 20} = \frac{17}{20}$.
4. L'aire du carré est égale à $200^2 = 40\,000$; l'aire de la cible est égale à $\pi \times 100^2 = 10\,000\pi$.
   La probabilité est donc égale à $\frac{10\,000\pi}{40\,000} = \frac{\pi}{4} \approx 0{,}785$, soit 0,79 au centième près.

Analyse

Considérations générales

Cet exercice propose à l'élève de réfléchir à une implémentation en Scratch d'une estimation de surface par une méthode probabiliste de type Monte-Carlo. Il y a très peu de questions concernant l'informatique : 3.(a), 3.(b) et 3.(c). L'exercice se concentre plutôt sur les calculs de mathématiques. Il met surtout l'accent sur l'aspect géométrique du problème (calcul de la distance du point F à l'origine) ainsi que sur le calcul de la fréquence à laquelle la cible est touchée. L'objectif de cette expérience n'est pas explicité dans l'énoncé. Le lien entre la probabilité de toucher la cible et la surface du disque gris est évoqué dans la question 4, mais pas ou peu exploité. L'élève n'est pas invité à comparer la fréquence empirique obtenue sur l'exemple et la fréquence théorique. De plus, nulle part il n'est mentionné que l'un des intérêts de cette expérience est d'estimer le nombre π en cherchant une approximation de l'aire du disque.

En ce qui concerne les attendus du cycle 4 en algorithmique et programmation, une part importante de l'activité de programmation de l'expérience est prise en charge par l'énoncé. L'exercice évalue notamment la capacité de l'élève à comprendre une boucle « répéter *n* fois ». Cependant la mise en œuvre des expressions arithmétiques et des instructions d'affectation demeurent à un niveau modeste. Plus ennuyeux, l'élève doit également disposer de connaissances spécifiques au logiciel Scratch :

— existence des variables spéciales abscisse x et ordonnée y,

— instruction aller à x : y :,

— expression nombre aléatoire entre et.

Comme dans la plupart des exercices de brevet étudiés ici, une connaissance spécifique de ce langage est donc évaluée.

Remarques sur la discrétisation du problème

Il est intéressant de noter que le bloc [nombre aléatoire entre -100 et 100] induit une discrétisation plutôt grossière du problème. En effet, les nombres renvoyés par cette expression sont des entiers. Cette caractéristique (ainsi que la non prise en compte des points appartenant au cercle lui-même) empêchent l'obtention d'une estimation très précise. Pour remédier à ce problème, on pourra par exemple tirer des nombres entiers plus grands et les diviser par une constante bien choisie pour se rapporter à l'intervalle de valeurs souhaité.

Expérience aléatoire versus recherche exhaustive

Plutôt que de tirer aléatoirement des points à coordonnées entières du plan, on pourrait d'ailleurs obtenir plus simplement une approximation de l'aire du disque en énumérant de manière exhaustive tous les points à coordonnées entières du carré blanc, et dénombrer ceux qui parmi eux appartiennent au disque. Dans le cas particulier où le tirage aléatoire est aussi grossier, une telle recherche exhaustive semble non seulement tout à fait justifiée mais aussi beaucoup plus efficace que l'utilisation d'une expérience aléatoire.

Si idéalement le pas entre deux points consécutifs dans la recherche exhaustive est de 1, il conduit à un temps d'exécution long. L'enseignant pourra faciliter la mise en évidence du côté discret de l'expérience et accélérer l'exécution en augmentant le pas entre deux points.

Le compteur : la variable [score]

La question 3.b travaille sur le compteur, ce qui est une bonne idée mais n'aurait-il pas fallu plutôt appeler la variable [score] juste [S] pour inciter les élèves à avoir une réflexion plus poussée. En effet, le nom de la variable donné par l'énoncé suffit à faire comprendre à l'élève de quoi il s'agit. Une question sur le renommage de cette variable permettrait alors de vérifier que l'élève a bien compris de quoi il s'agit. On pourrait par exemple demander " Quel nom explicite pourrait-on donner à la variable [S] ? " ou " Renommer la variable [S] pour faciliter la lecture de ce programme ".

Variables [carré de OF] et [distance]

Enfin, en ce qui concerne la rédaction du programme Scratch proposé, on peut s'interroger sur le choix du nom de la variable [carré de OF], qui fait référence à la figure, contrairement à la variable [distance] qui porte un nom générique.

Par ailleurs, il est particulièrement maladroit d'utiliser cette variable [distance] qui utilise la racine carrée et donc entraîne une comparaison de nombres flottants. En effet, nous connaissons les problèmes liés aux approximations des flottants dues à leur représentation en binaire. Il serait préférable de comparer le [carré de OF] au carré du rayon afin de rester dans les entiers et d'éviter entre autre des problèmes sur les cas limites.

Cette remarque qui peut sembler anodine tant que l'on travaille l'algorithme sur papier deviendra critique dès qu'on voudra implémenter le programme.

Attention, de plus, il faut garder en mémoire que pour les élèves de collège, la racine carrée n'est qu'une touche sur la calculatrice et qu'il ne la connaissent pas en tant que fonction. Il ne connaissent pas la notion de croissance de fonction, et ne sont pas en mesure de savoir que [distance < rayon] équivaut à [carré de OF < rayon * rayon].

De plus, la notion de distance dans le plan orthonormé ne figure pas dans les programmes avant la classe de seconde. Au collège, les élèves utilisent le théorème de Pythagore essentiellement dans le triangle rectangle, la notion de distance à l'origine en tant que racine de la somme des carrés des coordonnées n'est pas encore acquise par les élèves. L'énoncé présente donc une difficulté implicite dans les cas où une des coordonnées au moins n'est pas positive. C'est le triangle rectangle dessiné sur la figure de l'énoncé dans le quadrant supérieur droit qui les guide sur ce point.

Par contre c'est une bonne occasion d'aborder ces questions en classe et d'expliquer pourquoi il est préférable de travailler avec les entiers lors de la correction.

Propositions

Proposition 1 : en gardant l'expérience aléatoire

Cette proposition de sujet tient compte des remarques précédentes sur la variable score et du côté grossier de la discrétisation du problème.

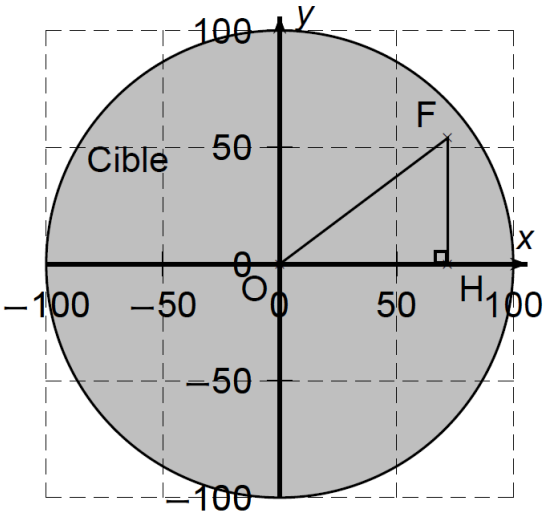
Dans tout l'exercice l'unité de longueur est le mm.

On lance une fléchette sur une plaque carrée sur laquelle figure une cible circulaire (en gris sur la figure). Si la pointe de la fléchette est sur le bord de la cible, on considère que la cible n'est pas atteinte.
On considère que cette expérience est aléatoire et l'on s'intéresse à la probabilité que la fléchette atteigne la cible.

— La longueur du côté de la plaque carrée est 200.

— Le rayon de la cible est 100.

— La fléchette est représentée par le point F de coordonnées $(x\ ;\ y)$ où x et y sont des nombres aléatoires compris entre -100 et 100.



1. Dans l'exemple ci-dessus, la fléchette F est située au point de coordonnées $(72\ ;\ 54)$.
   Montrer que la distance OF, entre la fléchette et l'origine du repère est 90.
2. D'une façon générale, quel nombre ne doit pas dépasser la distance OF pour que la fléchette atteigne la cible ?
3. On réalise un programme qui simule plusieurs fois le lancer de cette fléchette sur la plaque carrée et qui compte le nombre de lancers atteignant la cible. Le programmeur a créé trois variables nommées : carré de OF, distance et S.

```
1 quand [drapeau] est cliqué
2 mettre S à 0
3 répéter 120 fois
4     aller à x : (nombre aléatoire entre -10000 et 10000 / 100) y : (nombre aléatoire entre -10000 et 10000 / 100)
5     mettre carré de OF à (abscisse x * abscisse x + )
6     mettre distance à (racine de ( ))
7     si <distance < ( )> alors
8         ajouter 1 à S
```

(a) Lorsqu'on exécute ce programme, combien de lancers sont simulés ?

(b) Quel est le rôle de la variable S ? Renommer cette variable pour faciliter la lecture de ce programme.

(c) Compléter et recopier sur la copie uniquement les lignes 5, 6 et 7 du programme afin qu'il fonctionne correctement.

(d) Après une exécution du programme, la variable S est égale à 102.
À quelle fréquence la cible a-t-elle été atteinte dans cette simulation ?
Exprimer le résultat sous la forme d'une fraction irréductible.

4. On admet que la probabilité d'atteindre la cible est égale au quotient : aire de la cible divisée par aire de la plaque carrée.
Donner une valeur approchée de cette probabilité au centième près.

Correction de la proposition 1

Fichier Scratch correspondant à cette proposition : `sg_pondichery_2018_mai_proposition_1.sb3`.
programme en ligne

1. Le triangle OFH est rectangle en H ; du théorème de Pythagore appliqué à ce triangle on déduit : $OF^2 = OH^2 + HF^2$, soit $OF^2 = 72^2 + 54^2 = 5\,184 + 2\,916 = 8\,100$, donc $OF = \sqrt{8\,100} = 90$.
2. La fléchette doit être à l'intérieur du cercle, donc on doit avoir $OF^2 = x^2 + y^2 < 100^2$ ou encore $x^2 + y^2 < 10\,000$, x et y étant les coordonnées du point F.
3. (a) On simule 120 lancers.

   (b) S comptabilise le nombre de lancers ayant atteint la cible. On pourrait la renommer Score ou Compteur ou Nombre de succès etc.

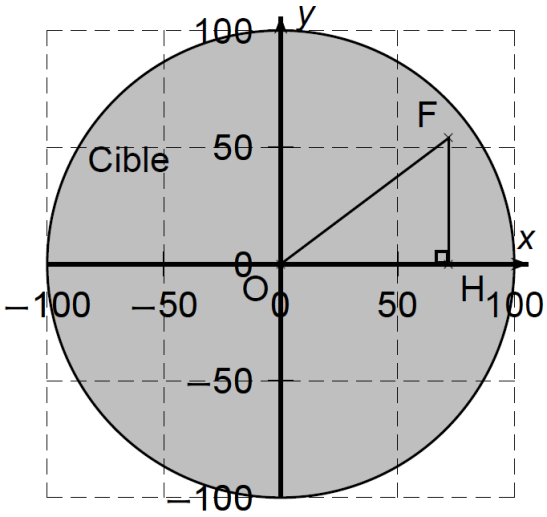
   (c) Dans la ligne 5, il faut compléter par ordonnée y * ordonnée y.
   Dans la ligne 6 il faut écrire carré de OF.
   Dans la ligne 7 il faut compléter avec le nombre 100.

   (d) Le nombre de réussites étant égal à 102 sur 120 lancers, la fréquence de réussite est égale à $\frac{102}{120} = \frac{51}{60} = \frac{3 \times 17}{3 \times 20} = \frac{17}{20}$.
4. L'aire du carré est égale à $200^2 = 40\,000$; l'aire de la cible est égale à $\pi \times 100^2 = 10\,000\pi$.
   La probabilité est donc égale à $\frac{10\,000\pi}{40\,000} = \frac{\pi}{4} \approx 0{,}785$, soit 0,79 au centième près.

Proposition 2 : Approximation de π avec une recherche exhaustive

Dans tout l'exercice l'unité de longueur est le mm.

On souhaite obtenir une approximation du nombre π en estimant le rapport entre l'aire d'un disque et de son carré exinscrit (c'est à dire un carré dans lequel de disque est inscrit). Pour estimer ce rapport, on parcourt les points à l'intérieur carré à intervalles réguliers et on regarde si ces points appartiennent ou non au disque. On calcule alors la fréquence des points du carré appartenant au disque.

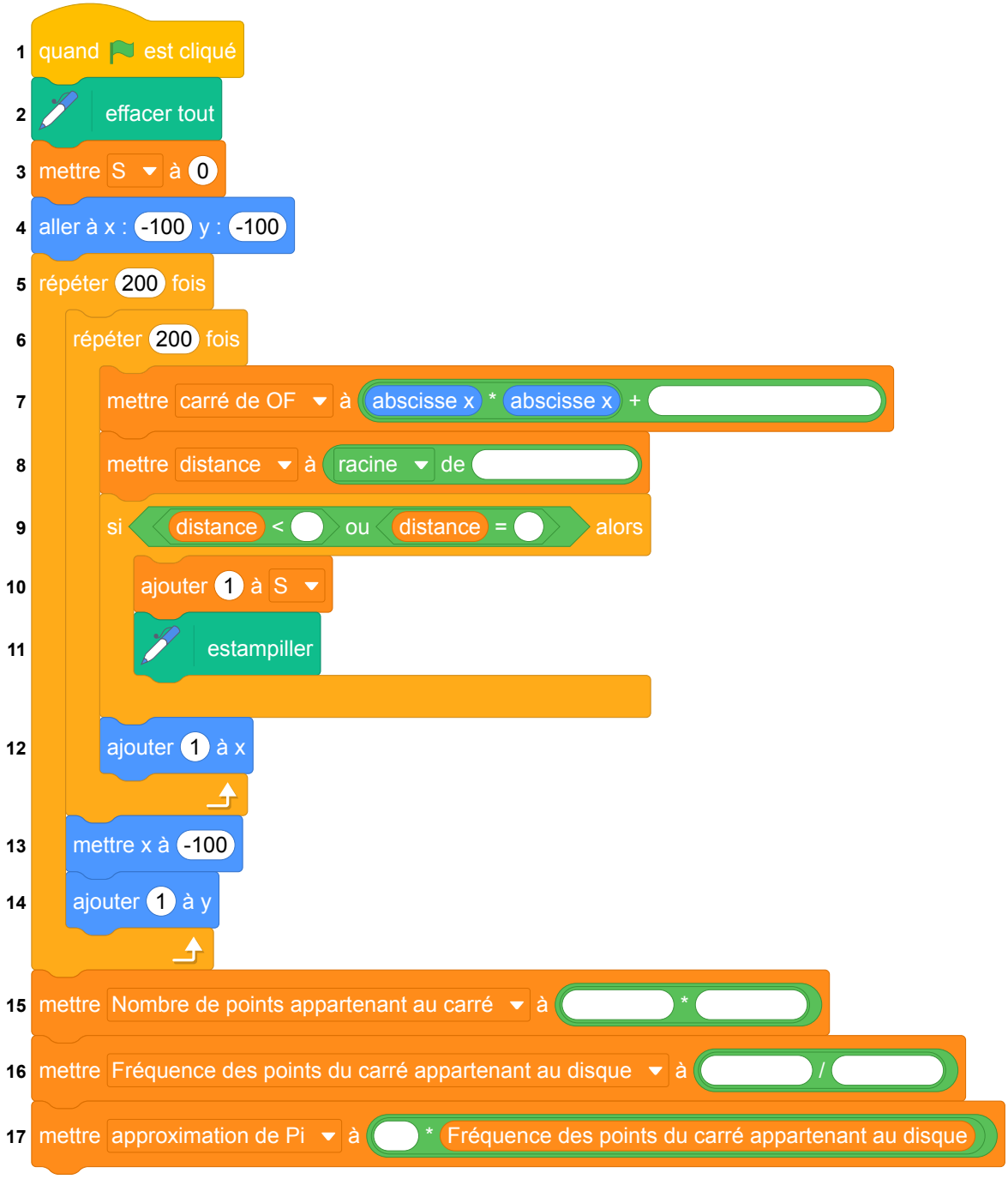


— La longueur du côté de la plaque carrée est 200.

— Le rayon de la cible est 100.

— Un point F de coordonnées $(x\ ;\ y)$ où x et y sont des nombres compris entre -100 et 100 appartient au carré.

1. Dans l'exemple ci-dessus, le point F est située au point de coordonnées $(72\ ;\ 54)$.
   Montrer que la distance OF, entre le point F et l'origine du repère est 90.
2. D'une façon générale, quel nombre ne doit pas dépasser la distance OF pour que le point appartienne au disque ?
3. Exprimer la distance OF en fonction des coordonnées de F : (x, y).
4. Exprimer l'aire d'un carré de côté $2R$.
5. Exprimer l'aire d'un disque de rayon R.
6. Calculer le rapport $q = \dfrac{\textit{Aire du disque}}{\textit{Aire du carre}}$.
7. Exprimer π en fonction du rapport q.
8. On réalise un programme qui parcourt les points du carré et compte le nombre de points appartenant au disque. Le programmeur a créé trois variables nommées : carré de OF, distance et S.



(a) Lorsqu'on exécute ce programme, combien de points sont parcourus au total ?

(b) Quel est le rôle de la variable S ? Renommer cette variable pour faciliter la lecture de ce programme.

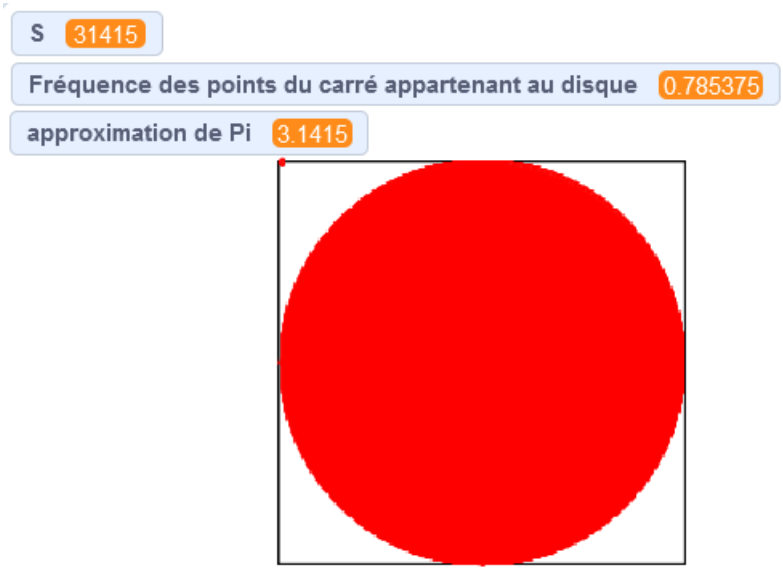
(c) Compléter et recopier sur la copie uniquement les lignes 7, 8, 9 et les 3 dernières lignes du programme afin qu'il fonctionne correctement.

(d) Après une exécution du programme, la variable S est égale à 31 415. Calculer la fréquence des points du carré appartenant au disque. Donner une valeur approchée de cette fréquence au centième près.

9. En déduire une approximation du nombre π au centième près.

Correction de la proposition 2

Cette version permet de visualiser le parcours du lutin et de visualiser le disque en estampillant le lutin quand il est dedans.



Fichier Scratch correspondant à cette proposition : `sg_pondichery_2018_mai_proposition_2.sb3`.
programme en ligne

1. Le triangle OFH est rectangle en H ; le théorème dePythagore appliqué ce triangle s'écrit : $\text{OF}^2 = \text{OH}^2 + \text{HF}^2$, soit $\text{OF}^2 = 72^2 + 54^2 = 5\,184 + 2\,916 = 8\,100$, donc $\text{OF} = \sqrt{8\,100} = 90$.
2. F appartient au disque de centre O et de rayon 100 donc $OF \leq 100$.
3. Le triangle OFH est rectangle en H ; du théorème de Pythagore appliqué à ce triangle on déduit : $\text{OF}^2 = \text{OH}^2 + \text{HF}^2$, soit $\text{OF}^2 = x^2 + y^2$.

   Le point F doit être à l'intérieur du disque, donc on doit avoir $\text{OF}^2 = x^2 + y^2 \leq 100^2$ ou encore $x^2 + y^2 \leq 10\,000$, x et y étant les coordonnées du point F.
4. aire du carré de côté $2R$= $(2R)^2 = 4R^2$
5. aire du disque de rayon R = πR^2
6. $q = \dfrac{\pi R^2}{4R^2} = \dfrac{\pi}{4}$
7. $\pi = 4q$
8. (a) Il y a deux boucles répéter imbriquées de 200 itérations chacune. Il y a donc $200 \times 200 = 40\,000$ points de parcourus.

   (b) La variable S est une variable compteur. Elle comptabilise le nombre de points parcourus appartenant au disque. On peut la renommer Score ou Compteur ou Nombre de succès.

(c) Programme complété :

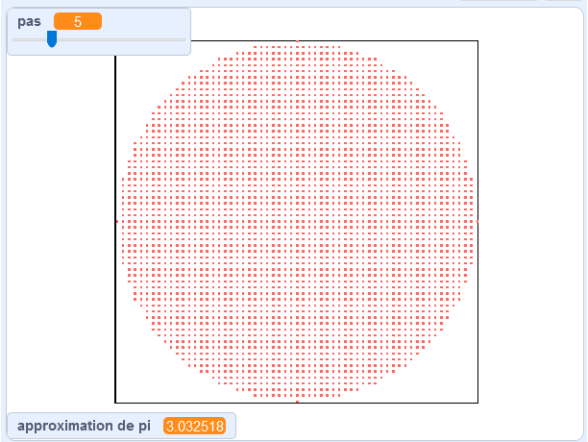
```
quand [drapeau] est cliqué
effacer tout
mettre S à 0
aller à x : -100 y : -100
répéter 200 fois
    répéter 200 fois
        mettre carré de OF à (abscisse x * abscisse x + abscisse y * abscisse y)
        mettre distance à (racine de carré de OF)
        si <(distance < 100) ou (distance = 100)> alors
            ajouter 1 à S
            estampiller
        ajouter 1 à x
    mettre x à -100
    ajouter 1 à y
mettre Nombre de points appartenant au carré à (200 * 200)
mettre Fréquence des points du carré appartenant au disque à (S / Nombre de points appartenant au carré)
mettre approximation de Pi à (4 * Fréquence des points du carré appartenant au disque)
```

(d) fréquence $= \dfrac{31\,415}{40\,000} \approx 0{,}79$

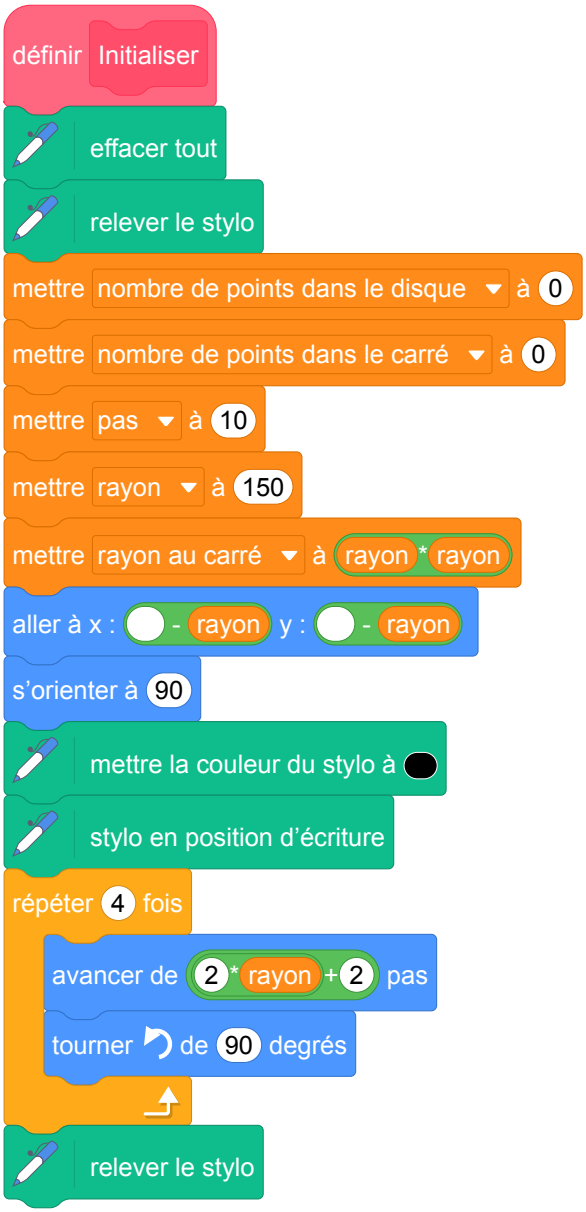
9. $\pi \approx 4 \times \dfrac{31\,415}{40\,000} \approx 3{,}14$

Proposition 3 : pour l'enseignant

Enfin voici une version du programme pour l'enseignant de l'approximation de π par recherche exhaustive des points du disque, qui tient compte du pas et de la question des flottants : programme en ligne sur
`https://scratch.mit.edu/projects/462877155/`.
Cette version peut être utilisée lors de la correction de l'exercice précédent proposé en classe de troisième pour aider les élèves à visualiser le côté discret de l'expérience.



Elle permet également d'aborder la question de l'approximation du calcul flottant et des risques encourus lors de l'usage de test d'égalité. Elle peut également donner lieu à un exercice en classe de seconde une fois la fonction racine carrée et la notion de croissance de fonction abordée.



```
définir Remplissage du disque
répéter jusqu'à ce que (ordonnée y > rayon)
    répéter jusqu'à ce que (abscisse x > rayon)
        stylo en position d'écriture
        ajouter 1 à nombre de points dans le carré
        mettre distance au carré à (abscisse x * abscisse x + ordonnée y * ordonnée y)
        si (rayon au carré < distance au carré) alors
            basculer sur le costume vert
        sinon
            basculer sur le costume rouge
            ajouter 1 à nombre de points dans le disque
            estampiller
        ajouter pas à x
    mettre x à ( - rayon)
    ajouter pas à y
```

```
définir Approximation
mettre approximation de pi à (4 * nombre de points dans le disque / nombre de points dans le carré)
```

4 Amérique du Nord, juin 2018, série générale

Références

Série : générale

Académie : Amérique du Nord

Date : juin 2018

Exercice 4 - 14 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géometrie
- agrandissement réduction
- déplacement
- grandeurs et mesures

Thèmes informatiques

- variable déclaree
- activité du lutin
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Amérique du Nord, juin 2018

Dans cet exercice, aucune justification n'est attendue.

Simon travaille sur un programme. Voici des copies de son écran :

Script principal

```
quand [drapeau] est cliqué
aller à x : -200 y : 0
s'orienter à 90
effacer tout
mettre la taille du stylo à 1
mettre côté à 40
répéter 4 fois
    carré
    avancer de côté pas
    ajouter 20 à côté
```

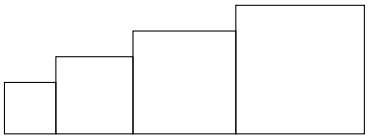
Bloc carré

```
définir carré
stylo en position d'écriture
répéter 4 fois
    avancer de côté pas
    tourner ↺ de 90 degrés
relever le stylo
```

Information

L'instruction `s'orienter à 90` signifie qu'on se dirige vers la droite.

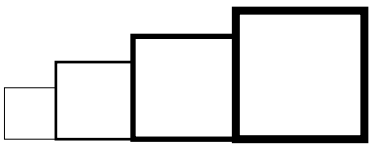
1. Il obtient le dessin ci-contre.



   (a) D'après le script principal, quelle est la longueur du côté du plus petit carré dessiné ?

   (b) D'après le script principal, quelle est la longueur du côté du plus grand carré dessiné ?

2. Dans le script principal, où peut-on insérer l'instruction `ajouter 2 à la taille du stylo` de façon à obtenir le dessin ci-contre ?

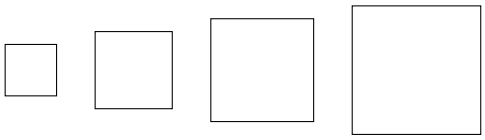


3. On modifie maintenant le script principal pour obtenir celui qui est présenté ci-contre :

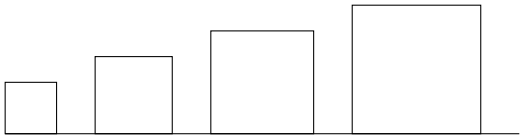
```
quand [drapeau] est cliqué
aller à x : -200 y : 0
s'orienter à 90
effacer tout
mettre la taille du stylo à 1
mettre côté à 40
répéter 4 fois
    carré
    avancer de côté + 30 pas
    ajouter 20 à côté
```

Parmi les dessins ci-dessous, lequel obtient-on ?

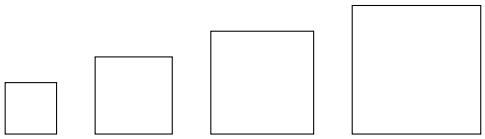
Dessin 1



Dessin 2



Dessin 3



Pour rappel : le bloc carré

Correction

Fichier scratch : `sg_amerique_du_nord_2018_juin.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/426280912`

1. (a) Au départ côté est mis à 40 ; le premier carré a ses côtés de longueur 40.

   (b) Dans la boucle du script principal, côté est augmenté 4 fois de 20, donc la valeur finale de la variable côté est $40 + 4 \times 20$. Cependant le dernier carré est dessiné avec l'avant dernière valeur de cette variable et donc a pour longueur de côtés :
   $40 + 3 \times 20 = 100$.

2. Il faut augmenter la taille du stylo à la fin de chaque tracé de carré, donc après l'instruction : ajouter 20 à côté

3. On obtient le dessin n° 3.

Analyse

Il s'agit d'un exercice de lecture, de compréhension et de modification de programme. Il traite du tracé de carrés de différentes tailles et fait appel à un bloc sans paramètre pour les tracer. Si nous pouvons comprendre que pour cette deuxième édition du brevet avec des exercices d'informatique il y a eu une volonté de ne pas utiliser de blocs à paramètres dans les sujets, nous trouvons dommage de ne pas profiter de l'occasion pour les travailler aujourd'hui avec nos élèves.

Nous pouvons par ailleurs nous demander jusqu'où il est utile de « préparer l'état initial »de Scratch ?

Nous pourrions par exemple relever le stylo au début du programme pour nous assurer de son état.

Les questions sont pertinentes. Elles permettent de vérifier si l'élève a compris comment évolue le contenu des variables, mais il serait intéressant d'approfondir les questions sur la boucle qui permet de répéter le tracé des carrés. Il suffirait pour cela d'ajouter des dessins avec la première question pour avoir une meilleure perception de la compréhension des élèves. Par exemple :

— Ajouter un dessin où il n'y a pas 4 carrés mais 3 ou 5 carrés.

— Ajouter un dessin où les carrés sont espacés d'écarts grandissants... Une question pourrait alors être ajoutée pour faire émerger la compréhension de l'instruction avancer de côté pas avant la première question de l'énoncé original en demandant aux élèves lequel des deux dessins correspond au programme initial.

La proposition d'énoncé qui se trouve ci-après tient compte de ces différentes remarques.

Propositions

Proposition d'énoncé

Dans cet exercice, aucune justification n'est attendue.

Simon travaille sur un programme. Voici des copies de son écran :

Script principal

```
1  quand [drapeau] est cliqué
2  relever le stylo
3  aller à x : -200 y : 0
4  s'orienter à 90
5  effacer tout
6  mettre la taille du stylo à 1
7  mettre côté à 40
8  répéter 4 fois
9      carré côté
10     avancer de côté pas
11     ajouter 20 à côté
```

Bloc carré

```
définir carré longueur du côté
stylo en position d'écriture
répéter 4 fois
    avancer de longueur du côté pas
    tourner ↺ de 90 degrés
relever le stylo
```

Information

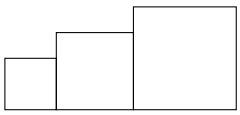
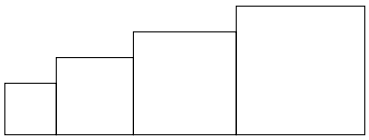
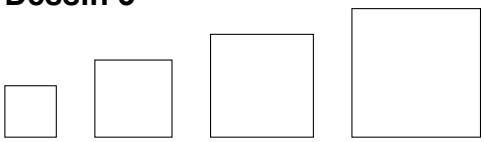
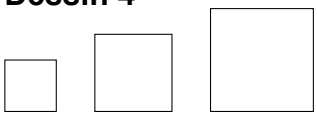
L'instruction `s'orienter à 90` signifie qu'on se dirige vers la droite.

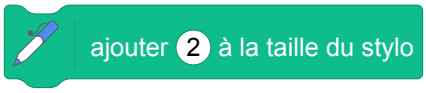
1. Il obtient un des 4 dessins ci-dessous.

   (a) Lequel de ces 4 dessins correspond au script proposé ?

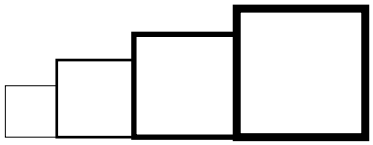
   (b) D'après le script principal, quelle est la longueur du côté du plus petit carré dessiné ?

   (c) D'après le script principal, quelle est la longueur du côté du plus grand carré dessiné ?

Dessin 1 	**Dessin 2** 
Dessin 3 	**Dessin 4** 

2. Dans le script principal, où peut-on insérer l'instruction 

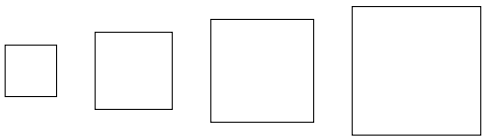
de façon à obtenir le dessin ci-dessous ?



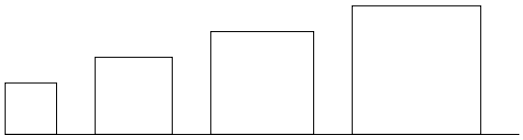
3. On modifie maintenant le script principal pour obtenir celui qui est présenté ci-contre :

Parmi les dessins ci-dessous, lequel obtient-on ?

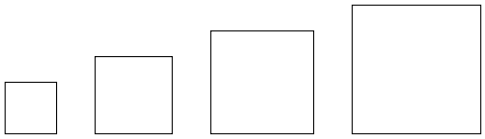
Dessin 1



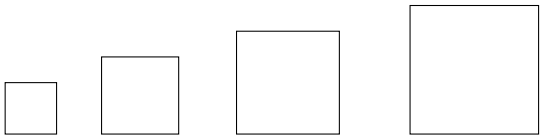
Dessin 2



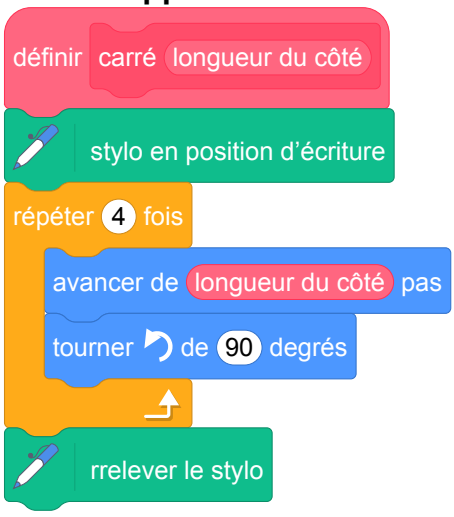
Dessin 3



Dessin 4



Pour rappel : le bloc carré



5 Asie, juin 2018, série générale

Références

Série : générale

Académie : Asie

Date : juin 2018

Exercice 4 - 12 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- agrandissement réduction
- rosace
- déplacement
- périmètre

Thèmes informatiques

- activité du lutin
- itération bornée

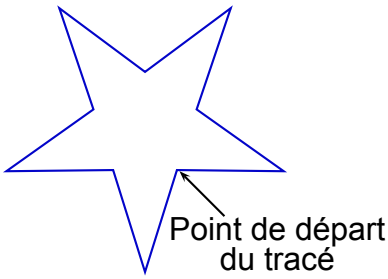
Compétences travaillées

- comprendre un algorithme
- compléter un algorithme
- modifier un algorithme

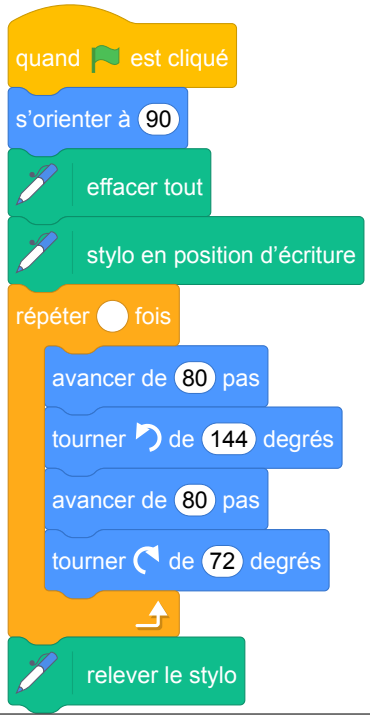
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Asie, juin 2018

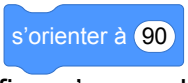
Arthur doit écrire un programme avec Scratch pour dessiner une étoile comme le dessin représenté ci-contre.
Il manque dans son programme le nombre de répétitions.



Programme commencé par Arthur

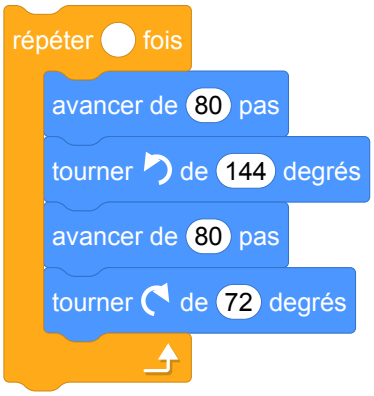


Information
L'instruction



signifie qu'on se dirige vers la droite.

1. Quel nombre doit-il saisir dans la boucle « répéter » pour obtenir l'étoile ?
2. Déterminer le périmètre de cette étoile.
3. Arthur souhaite agrandir cette étoile pour obtenir une étoile dont le périmètre serait le double, en modifiant son programme.
   Recopier la partie du programme ci-contre sur la copie en modifiant les valeurs nécessaires pour obtenir cette nouvelle étoile.



Correction

Fichier scratch : `sg_asie_2018_juin.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/408264270`

1. Il faut répéter le motif 5 fois.
2. Pour chaque motif on avance de $80 + 80 = 160$, donc le périmètre de la figure est égal à $5 \times 160 = 800$.
3. Pour avoir un périmètre double il suffit de remplacer les deux « 80 » par « 160 ».

Analyse

L'exercice est plus mathématiques qu'informatique. Il nécessite de lire un programme mais il traite plutôt de notions mathématiques : agrandissement, réduction, périmètre... Côté informatique, il y a une question sur la boucle mais cela reste assez pauvre.

Pour la question 2, il serait préférable d'indiquer dans quelle unité exprimer le périmètre. En Scratch 2 (énoncé original) on pourrait parler d'unité Scratch, et en Scratch 3, on pourrait préciser « en nombre de pas » afin que les élèves ne cherchent pas à prendre de mesures sur la figure et tirent bien les informations du programme.

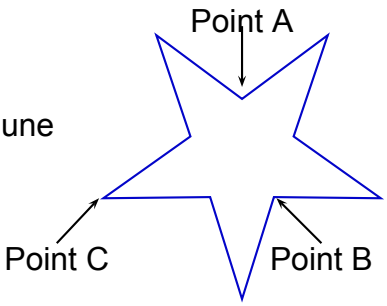
On pourrait retirer l'indication sur le point de départ de la figure du schéma. En effet, celle-ci n'apporte rien à l'exercice. Cela permettrait de poser des questions supplémentaires sur le point de départ du tracé et sur le motif de base construit par le bloc répéter et donc d'insister sur la compréhension du programme par les élèves.

Différentes réponses sont possibles pour doubler le périmètre : remplacer 80 par 160 ou remplacer 80 par 80×2 ou encore ajouter une variable côté égale à 80 et doubler sa valeur, etc.

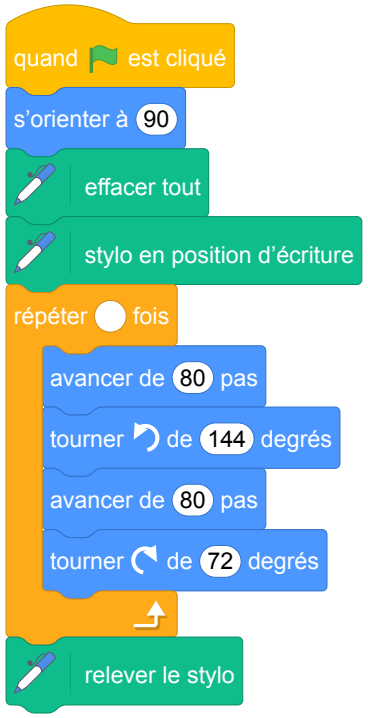
Propositions

Proposition d'énoncé

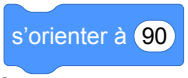
Arthur doit écrire un programme avec Scratch pour dessiner une étoile comme le dessin représenté ci-contre.
Il manque dans son programme le nombre de répétitions.



Programme commencé par Arthur

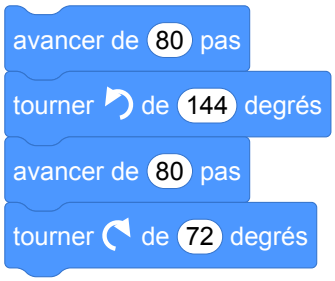


Information
L'instruction



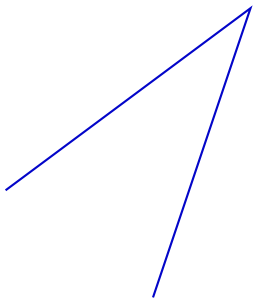
signifie qu'on se dirige vers la droite.

1. Quel nombre doit-il saisir dans la boucle « répéter » pour obtenir l'étoile ?
2. Le bloc d'instruction se trouvant dans la boucle est :

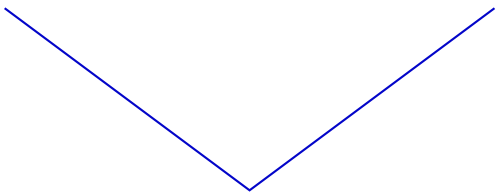


Parmi les deux schémas suivants, lequel est celui qui est dessiné lors de l'exécution de ce bloc ? Justifier la réponse.

Dessin n° 1



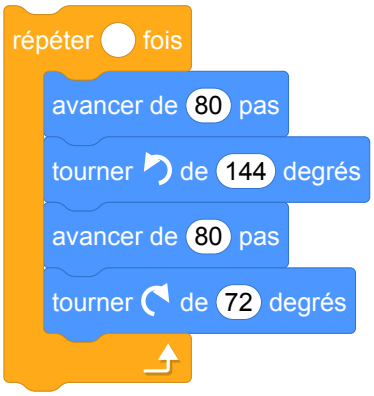
Dessin n° 2



3. Parmi les points A, B et C, quel est le point de départ du tracé ? Justifier la réponse. (Attention à bien prendre en compte l'orientation de départ du lutin !)
4. Déterminer le périmètre de cette étoile en nombre de pas.

5. Arthur souhaite agrandir cette étoile pour obtenir une étoile dont le périmètre serait le double, en modifiant son programme.

   Recopier la partie du programme ci-contre sur la copie en modifiant les valeurs nécessaires pour obtenir cette nouvelle étoile.



Correction de la proposition précédente

1. Il faut répéter le motif 5 fois.
2. Il s'agit du dessin n°1. En tournant de 144°, le lutin dessine un angle supplémentaire à 144° c'est-à-dire un angle aigu de 36°.
3. Le point de départ est le point B. En effet, cela ne peut être le point A à cause de l'orientation initiale du lutin vers la droite. Comme après avoir tracé le premier segment, le lutin tourne à gauche de 144°, le point de départ est le point B.
4. Pour chaque motif on avance de $80 + 80 = 160$ pas, donc le périmètre de la figure est égal à $5 \times 160 = 800$ pas.
5. Pour avoir un périmètre double il suffit de remplacer les deux « 80 » par « 160 ».

6 Centres Étrangers, juin 2018, série générale

Références

Série : générale

Académie : Centres Étrangers

Date : juin 2018

Exercice 6 - 18 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- déplacement

Thèmes informatiques

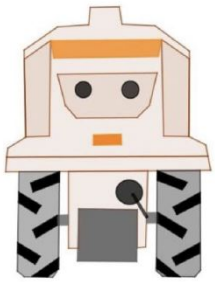
- scène
- itération bornee
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme
- écrire un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Centres Étrangers, juin 2018



Le maraîchage est l'activité professionnelle qui consiste à cultiver les légumes, certains fruits, fleurs ou plantes aromatiques.
Afin de diminuer la pénibilité des travaux de maraîchage, un agriculteur a acquis un robot électrique pour effectuer le désherbage de ses cultures

Partie A. Parcours du robot

Le robot doit parcourir 49 allées parallèles écartés de 1 m, représentées sur le schéma ci-dessous.

Les 48 premières allées, situées dans une parcelle rectangulaire, mesurent 80 m de long :

- la 1re allée est $[PQ]$;
- la 2e allée est $[RS]$;
- la 3e allée est $[TU]$;
- les allées 4 à 47 ne sont pas représentées ;
- la 48e allée est $[CB]$.

la 49e (dernière allée) $[DE]$ est située dans une parcelle triangulaire.

Montrer que la longueur de la dernière allée est $DE = 64$ m.

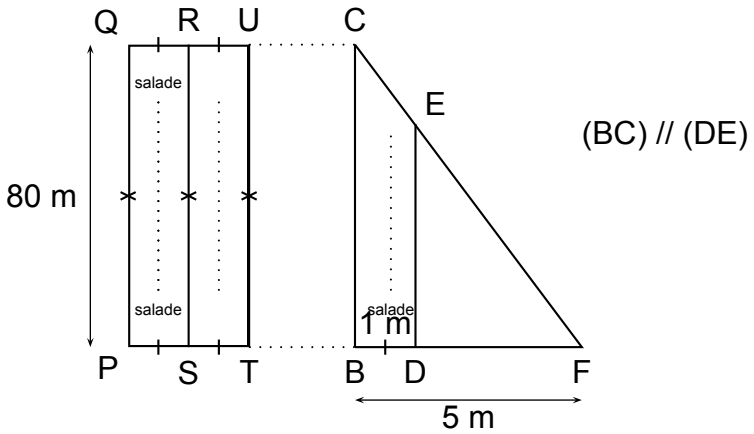


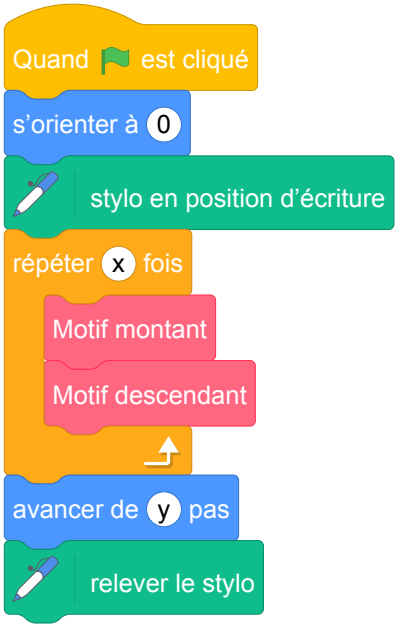
Schéma 2 du terrain non à l'échelle : vue de dessus

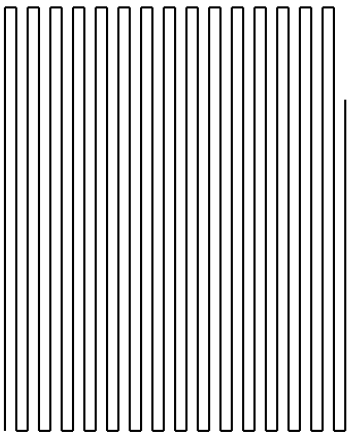
Partie B. Programme de déplacement du robot

On souhaite programmer le déplacement du robot du point P au point E. Le script ci-dessous, réalisé sous Scratch, est incomplet. Toutes les allées sont parcourues une seule fois. L'image « Robot » correspond au résultat attendu lorsque le drapeau vert est cliqué.

On rappelle que l'instruction

signifie que le robot se dirige vers le haut.





Script incomplet de déplacement du robot

Image à obtenir avec le script complet

Pour répondre aux questions 1 et 2, utiliser autant que nécessaire les blocs :

1. Le nouveau bloc « Motif montant » doit reproduire un déplacement du type *P*-*Q*-*R* (voir schéma 2) et positionner le robot prêt à réaliser le motif suivant. Écrire une succession de 4 blocs permettant de définir : « Motif montant ».
2. Le nouveau bloc « Motif descendant » doit reproduire un déplacement du type R-S-T (voir schéma 2) et positionner le robot prêt à réaliser le motif suivant. Quelle(s) modification(s) suffit-il d'apporter au bloc « Motif montant » pour obtenir le bloc « Motif descendant » ?
3. Quelles valeurs faut-il donner à *x* et à *y* dans le script principal pour que le programme de déplacement du robot donne le résultat attendu.

Correction

Fichiers Scratch :

- `sg_centres_etrangers_2018_juin.sb3`
- `sg_centres_etrangers_2018_juin_correction_v1.sb3`
- `sg_centres_etrangers_2018_juin_correction_v2.sb3`

Liens vers les fichiers Scratch sur le Studio Scratch c3i-brevet :

- `https://scratch.mit.edu/projects/386266499/`
- `https://scratch.mit.edu/projects/386273317/`
- `https://scratch.mit.edu/projects/386274723/`

La version 2 de la correction, comporte un cœfficient d'agrandissement des longueurs pour que le parcours du robot soit visible.

Partie A. Parcours du robot

On sait que (CE) et (BD) se coupent en F et que $(BC)//(DE)$.

D'après la propriété de Thalès, on a : $\frac{FD}{FB} = \frac{DE}{BC} = \frac{FE}{FC}$. Soit $\frac{4}{5} = \frac{DE}{80} = \frac{FE}{FC}$.

Donc DE $= \frac{80 \times 4}{5} = 64$ m.

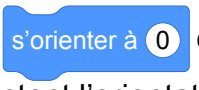
Partie B. Programme de déplacement du robot

1.

2. Il suffit de tourner dans l'autre sens.
3. $\dfrac{48}{2} = 24$ donc $x = 24$. $y = 64$ (dernière longueur).

Analyse

La partie A de l'exercice permet de comprendre le circuit du robot et traite le théorème de Thalès. La partie B de l'exercice traite le programme qui définit le déplacement du robot.

Le sens de l'instruction 

est donné car cela relève d'une connaissance du langage Scratch.

Les angles qui représentent l'orientation du lutin (0° vers le nord, 90° vers l'est, 180° vers le sud, −90° vers l'ouest) ne sont pas cohérents avec le cercle trigonométrique (0° vers l'est, 90° vers le nord, 180° vers l'ouest, −90° vers le sud). C'est un point à prendre en compte car au lycée, quand les élèves étudieront le cercle trigonométrique, cela peut devenir une source de confusion. Les autres instructions sont sensées d'être connues par le candidat, y compris les instructions de l'extension stylo. Cependant, le document d'accompagnement sur l'algorithmique et la programmation diffusé par l'Éduscol (voir [6]) insiste sur le fait qu'il faut préciser les conventions de repérage de Scratch i.e. les coordonnées de la scène de Scratch ainsi que les angles utilisés pour les directions définies sur cette scène.

Pour répondre aux trois questions de la partie B, il faut connaître l'échelle qui sera utilisée par le programme Scratch. Pourtant cette question d'échelle est ignorée par l'exercice. Il est important de préciser, par exemple, que 1m sera représenté par 1 pas du lutin donc du robot dans cet exercice. Le nombre y demandé dans la question 3 dépend de l'échelle utilisée par le robot or celle-ci n'est pas donnée dans l'exercice.

La première question de la partie B n'est pas claire car il n'y a aucune information exacte donnée sur l'orientation du robot à la fin du motif montant. La seule précision est qu'il faut positionner le robot prêt à réaliser le motif suivant. Or quoique ce soit l'orientation du robot à la fin du motif montant, le motif descendant peut l'adapter. L'état du robot au début et à la fin d'un motif est une information indispensable pour la programmation. Dans certains cas, le motif montant n'est pas obligé de contenir 4 blocs. Les motifs suivants répondent aux deux premières questions :

En fait, dans l'exercice, l'état du lutin est présupposé, pas trop haut, pas trop à droite, en réalité à $(0, 0)$ de la scène qui est supposé également vide au début de l'exécution. Dans les projets informatiques, ce problème d'état dans un programme est sous la responsabilité du programmeur qui doit respecter un cahier de charge. Dans cet exercice, les blocs sont insuffisamment spécifiés pour en dire plus.

D'autre part, le candidat peut très bien utiliser le bloc s'orienter à 180 pour orienter le robot vers le bas, ceci n'est pas interdit mais l'instruction n'est pas proposée dans l'exercice.

Propositions

Les propositions qui suivent concernent la partie B de l'exercice.

Il est possible de définir l'orientation exacte du robot au début et à la fin de chaque motif avant de demander les blocs qui les définissent. Il est donc important d'insister sur l'état du lutin à ces moments-là. Par exemple, au début du motif montant, le robot est orienté vers le haut, ceci est donné dans l'algorithme par le bloc s'orienter à 0. Et surtout, on peut expliciter qu'à la fin du motif montant, le robot doit être orienté vers le bas. Nous pouvons imaginer d'écrire un bloc qui dessine le trajet indépendamment de l'état du lutin. Le contrat du programmeur dans ce cas se limiterait à dessiner le trajet du robot quelque soit l'état initial du lutin.

La deuxième question n'exige pas l'écriture des blocs du motif descendant. Selon la réponse donnée à la question 1, cette question peut être confuse. Il vaut mieux demander les blocs de ce motif descendant pour éviter toutes confusions.

Si on utilise l'échelle 1 m vaut 1 pas dans le programme Scratch (voir le fihier Scratch correction version 1), le motif dessiné sur la scène est illisible. Un pas étant très petit, on ne distingue pas les allées que le robot est censé de parcourir. Cela est dommage parce que la différence entre un objet et sa représentation est un point intéressant d'étude. Nous pouvons travailler sur les échelles en définissant un cœfficient de proportionnalité en tant que variable (voir le fichier Scratch correction version 2). Toutes les longueurs seront donc multipliées par ce cœfficient pour agrandir le parcours. Dans ce cas, l'emplacement du lutin sur la scène devient important car pour voir l'ensemble des allées, il faut déplacer le début des allées i.e. il faut bouger le lutin. Nous pouvons alors, par exemple, ajouter l'instruction aller à x : 0 y : -100 pour un cœfficient égal à 3.

Un travail intéressant à poursuivre en classe serait d'intégrer une photo satellite sur laquelle on voit un champ et le trajet éventuel du robot. L'élève programmerait alors le robot pour qu'il traverse les allées déjà existantes. Cette proposition nécessite obligatoirement l'explicitation de la question de l'échelle.

7 Métropole, juin 2018, série générale

Références

Série : générale

Académie : Métropole

Date : juin 2018

Exercice 6 - 16 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- agrandissement réduction
- déplacement
- homothétie

Thèmes informatiques

- scène
- variable déclarée
- bloc sans paramètres

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole, juin 2018

Les longueurs sont en pas.

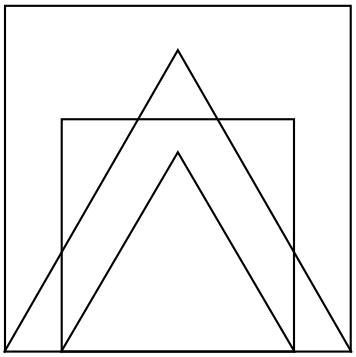
L'expression s'orienter à 90 signifie que l'on s'oriente vers la droite.

On donne le programme suivant :

```
1  quand [drapeau] est cliqué
2  aller à x : 0 y : 0
3  stylo en position d'écriture
4  s'orienter à 90
5  mettre Longueur à 300
6  Carré
7  Triangle
8  avancer de Longueur / 6
9  mettre Longueur à ( )
10 Carré
11 Triangle
```

```
définir Carré
répéter 4 fois
    avancer de Longueur pas
    tourner ↺ de 90 degrés

définir Triangle
répéter 3 fois
    avancer de Longueur pas
    tourner ↺ de 120 degrés
```

1. On prend comme échelle 1 cm pour 50 pas.
   (a) Représenter sur votre copie la figure obtenue si le programme est exécuté jusqu'à la ligne 7 comprise.
   (b) Quelles sont les coordonnées du stylo après l'exécution de la ligne 8 ?
2. On exécute le programme complet et on obtient la figure ci-dessous qui possède un axe de symétrie vertical.



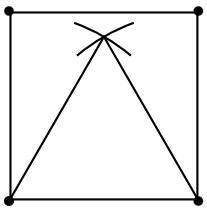
   Recopier et compléter la ligne 9 du programme pour obtenir cette figure.
3. (a) Parmi les transformations suivantes, translation, homothétie, rotation, symétrie axiale, quelle est la transformation géométrique qui permet d'obtenir le petit carré à partir du grand carré ? Préciser le rapport de réduction.
   (b) Quel est le rapport des aires entre les deux carrés dessinés ?

Correction

Fichier Scratch : `sg_metropole_2018_juin.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/772400390/`

1. (a) Voici la figure qui doit être tracée.



   (b) Après l'exécution de la ligne 8, le stylo sera à $x = 50$ et $y = 0$.

2. Pour tracer la figure intérieure on doit se décaler de 50 de chaque côté. Donc le côté intérieur sera de $300 - 2 \times 50 = 200$.

3. (a) Il s'agit d'une homothétie de rapport : $\dfrac{200}{300} = \dfrac{2}{3}$.

   (b) Par définition, si k est le rapport de réduction des longueurs, k^2 sera le rapport de réduction pour les aires. Donc : $k^2 = \left(\dfrac{2}{3}\right)^2 = \dfrac{4}{9}$.

Analyse

L'exercice original étant écrit en Scratch 2.0, le programme donné utilisait des pixels pour se déplacer. Ce document étant écrit sur Scratch 3.0, les pixels ont été transformés en pas.

Le programme fournit seulement une partie des instructions décrivant l'état initial du lutin. Il donne les blocs aller à x : 0 y : 0 et s'orienter à 90 alors que ceux-ci font partie de l'état par défaut de Scratch.

Il ne donne pas par contre, l'instruction effacer tout qui peut être vue comme un élément de l'état initial. Nous pensons que l'instruction d'aller à $(0, 0)$ est donnée car la question 1.(b) demande les coordonnées du lutin après l'exécution. Le stylo n'est pas relevé à la fin du programme donc le lutin n'est pas ramené à son état initial.

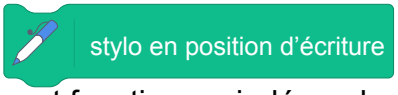
Les suppositions faites sur les états initiaux de Scratch et les instructions fournies ne sont pas toujours les mêmes d'un brevet à l'autre. L'hétérogénéité des contrats implicites sur les états initiaux et finaux est à remarquer entre les différents sujets. Une homogénisation de cette question peut enlever des confusions.

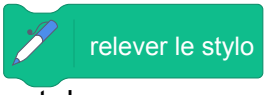
Propositions

Les blocs définir Carré et définir Triangle donnés dans l'exercice utilisent la variable Longueur qui est une variable globale. Nous pouvons étendre cet exercice en intégrant des blocs avec paramètre où la variable Longueur deviendrait le paramètre des blocs définir Carré Longueur et définir Triangle Longueur .

Les blocs de création de carré et de triangle peuvent être généralisés en utilisant qu'un seul bloc définir Polygone régulier Longueur Nombre de côtés qui dessinerait les figures en fonction du nombre de côtés fourni aussi en paramètre.

Une phrase explicitant l'état initial peut lever les éventuelles ambiguïtés sur ces questions d'état : "On suppose que le lutin est initialement positionné à $(0, 0)$, orienté vers la droite (ou vers l'est), que la scène est entièrement vide et que le stylo est levé.". Sauf qu'une telle phrase est trop compliquée pour être mise dans un sujet, elle peut perturber les élèves qui ne comprendront pas son utilité. Il serait préférable d'adopter un document d'accompagnement (ou enrichir l'existant, voir [6]) pour définir des consignes officielles.

Il aurait été plus judicieux de mettre les commandes 

et 

dans les blocs de carré et de triangle pour qu'ils puissent fonctionner indépendamment du programme principal. En plus, en l'état, le lutin écrit plusieurs fois sur le même trait en particulier pour se positionner.

Un travail à suivre en classe serait d'écrire un programme qui répète plusieurs fois les figures de base de plus en plus petites ou de plus en plus grandes. Ceci permettrait de travailler encore sur l'homothétie mais aussi sur la boucle "répéter".

8 Métropole, septembre 2018, série générale

Références

Série : générale

Académie : Métropole

Date : septembre 2018

Exercice 5 - 15 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- agrandissement réduction
- repérage
- déplacement
- calcul sans programme
- grandeurs et mesures
- homothétie

Thèmes informatiques

- variable déclaree
- activité du lutin
- calcul
- itération bornée
- bloc sans paramètre

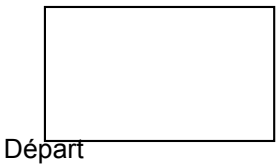
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	1
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole, septembre 2018

Sam a écrit le programme ci-dessous qui permet de tracer un rectangle comme ci-contre.
Ce programme comporte deux variables Longueur et Largeur qui représentent les dimensions du rectangle.

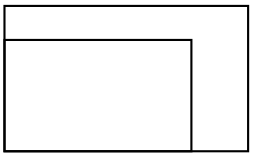


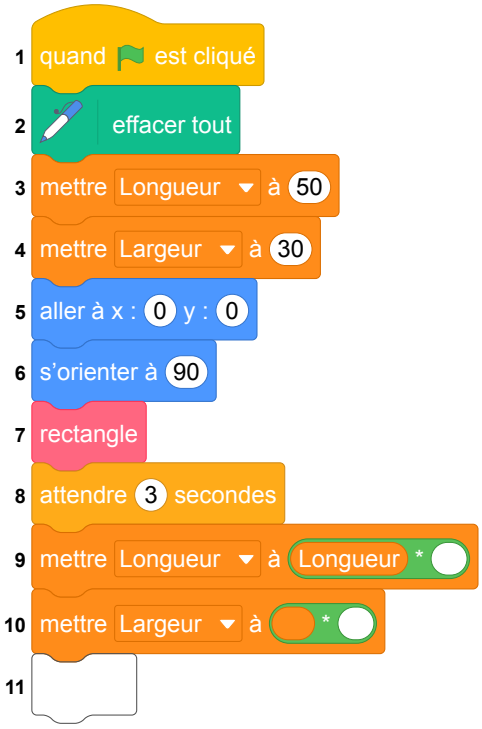
On rappelle que l'instruction s'orienter à 90 signifie que l'on s'oriente vers la droite.

Script	bloc rectangle
quand est cliqué effacer tout mettre Longueur à 50 mettre Largeur à 30 aller à x : 0 y : 0 s'orienter à 90 rectangle	définir rectangle stylo en position d'écriture répéter fois avancer de pas tourner de degrés avancer de pas tourner de degrés

1. Compléter le bloc rectangle ci-dessus avec des nombres et des variables pour que le script fonctionne.
   On recopiera et on complétera uniquement la boucle répéter sur sa copie.
2. Lorsque l'on exécute le programme, quelles sont les coordonnées du point d'arrivée et dans quelle direction est-on orienté ?
3. Sam a modifié son script pour tracer également l'image du rectangle par l'homothétie de centre le point de coordonnées (0 ; 0) et de rapport 1,3.

   (a) Compléter le nouveau script de Sam donné ci-contre afin d'obtenir la figure ci-dessous. On recopiera et on complètera sur sa copie les lignes 9 et 10 ainsi que l'instruction manquante en ligne 11.





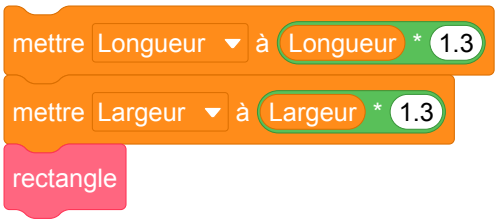
   b. Sam exécute son script. Quelles sont les nouvelles valeurs des variables Longueur et Largeur à la fin de l'exécution du script ?

Correction

Fichier Scratch : `sg_metropole_2018_septembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/446579265/`

1.

2. Les coordonnées sont celles du point de départ $(0, 0)$ et l'orientation à 90°.
3. (a)


   (b) À la fin de l'exécution du programme la longueur est de $50 \times 1,3 = 65$ et la largeur est de $30 \times 1,3 = 39$ pas.

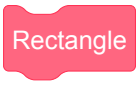
Analyse

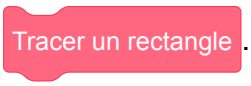
Cet exercice permet de travailler plusieurs compétences informatiques :

- compléter un programme ;
- comprendre un programme ;
- modifier un programme.

En mathématiques, il fait appel aux connaissances sur le rectangle et sur un agrandissement homothétique. Il est donc souhaitable de le proposer après avoir traité ce chapitre en quatrième ou en troisième afin de pouvoir faire des prolongements ou de poser une question explicite sur le sujet. Lors d'un exercice d'entrainement en classe les élèves pourraient reconnaître la transformation et calculer eux même le rapport de l'homothétie.

L'exercice est très guidé. Il se prête bien à un traitement sur papier et dans la forme originale, il peut être intéressant de le proposer lors d'une première évaluation. Une deuxième version, plus ouverte et adaptée pour un travail en classe est proposé plus loin.

Comme dans la plupart des sujets, le bloc est nommé par l'objet qu'il dessine 

mais il serait préférable de le nommer par son action 

.

Si l'exercice est donné en entraînement en classe ou lors de la correction d'un contrôle, on peut expliciter de deviner les coordonnées et l'orientation puis de les vérifier en programmant.

Nous regrettons encore une fois l'usage de blocs sans paramètres !

Dans la dernière partie, lignes 9 et 10, il aurait été préférable de laisser ... à la place du fois dans les opérateurs pour s'assurer que les élèves ont bien reconnu la situation de proportionnalité.

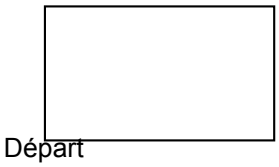
Attention également à la position du stylo car le bloc personnalisé qui trace un rectangle ne relève pas le stylo à la fin. En cas d'utilisations successives des traits non désirés seront tracés comme par exemple si on relance le programme principal qui efface tout avant de placer le lutin aux coordonnées initiales.

Propositions

Énoncé modifié d'après Métropole, septembre 2018

Sam a écrit le programme ci-dessous qui permet de tracer un rectangle comme ci-contre.
Ce programme comporte deux variables Longueur et Largeur qui représentent les dimensions du rectangle.

On rappelle que l'instruction s'orienter à 90 signifie que l'on s'oriente vers la droite.

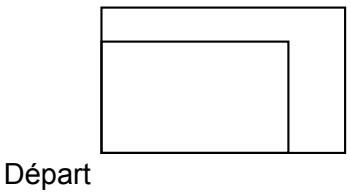


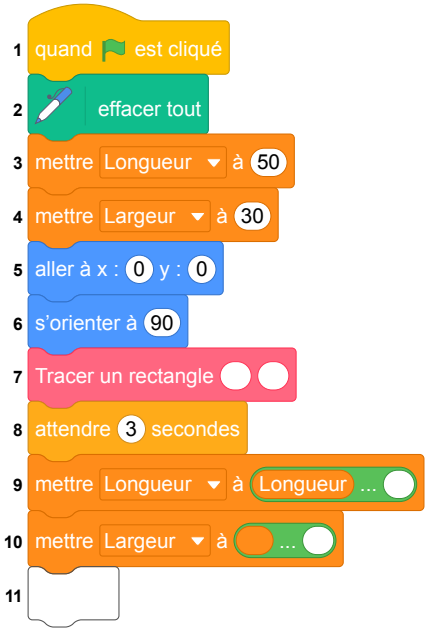
Script	bloc Tracer un rectangle
quand est cliqué effacer tout mettre Longueur à 50 mettre Largeur à 30 aller à x : 0 y : 0 s'orienter à 90 Tracer un rectangle	définir Tracer un rectangle côté 1 côté 2 stylo en position d'écriture répéter fois avancer de pas tourner de degrés avancer de pas tourner de degrés

1. Compléter le bloc Tracer un rectangle ci-dessus avec des nombres et des variables pour que le script fonctionne.
   On recopiera et on complétera uniquement la boucle répéter sur sa copie.
2. Compléter la dernière ligne du script principal (seule cette ligne est à recopier sur la copie et à compléter).
3. Deviner quelles sont les coordonnées du point d'arrivée et dans quelle direction est orienté le lutin à la fin de l'exécution de ce programme. Vous justifierez votre réponse.
4. Sam a modifié son script pour tracer également l'image du rectangle par une transformation.

   (a) Déterminer la transformation et ses caractéristiques.

   (b) Compléter le nouveau script de Sam donné ci-contre afin d'obtenir la figure ci-dessous. On recopiera et on complètera sur sa copie les lignes 9 et 10 en n'oubliant pas de remplacer ... par l'opérateur utilisé. Vous indiquerez également l'instruction manquante en ligne 11.



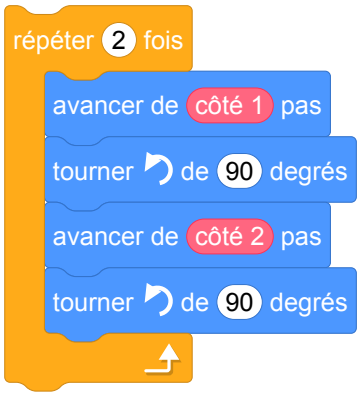


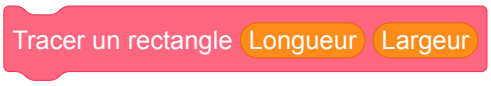
   (c) Sam exécute son script. Quelles sont les nouvelles valeurs des variables Longueur et Largeur à la fin de l'exécution du script ?

Correction de la version modifiée

Fichier Scratch : `sg_metropole_2018_septembre_modifie.sb3`

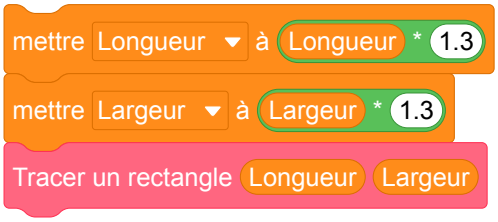
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/540804983/`

1. 

2. 

3. Les coordonnées sont celles du point de départ $(0, 0)$ et l'orientation à 90°.

4. (a) Il s'agit d'une homothétie ayant pour centre le point de départ et de rapport 1,3 (rapport d'agrandissement du rectangle).

   (b) 

   (c) À la fin de l'exécution du programme la longueur est de $50 \times 1,3 = 65$ et la largeur est de $30 \times 1,3 = 39$ pas.

9 Amérique du sud, novembre 2018, série générale

Références

Série : générale

Académie : Amérique du sud

Date : novembre 2018

Exercice 6 - 12 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- frise
- grandeurs et mesures
- translation

Thèmes informatiques

- activité du lutin
- itération bornée
- bloc sans paramètre
- bloc avec paramètres

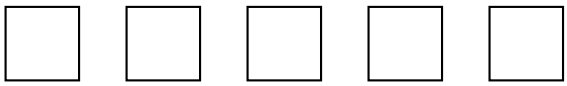
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

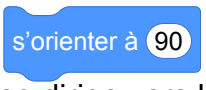
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Amérique du sud, novembre 2018

Léna et Youri travaillent sur un programme. Ils ont obtenu le dessin suivant :



Ils ont ensuite effacé une donnée par erreur dans le script principal.
Voici les copies d'écran de leur travail :

Programme		Pour information
Script principal	Bloc du Motif	
		L'instruction  signifie qu'on se dirige vers la droite.

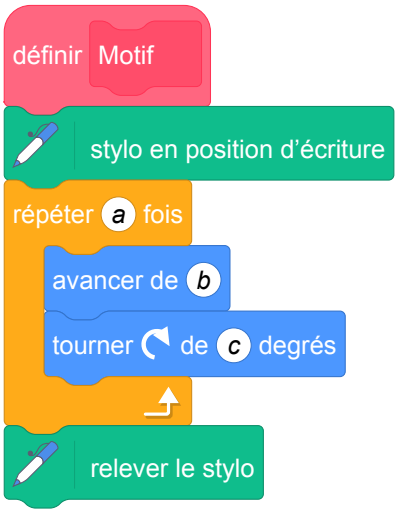
Valeur effacée

Dans cet exercice, aucune justification n'est demandée.

1. (a) La valeur effacée dans le script principal était-elle 40 ou bien 60 ?
   (b) Dessiner sur la copie ce qu'on aurait obtenu avec l'autre valeur.
   On représentera l'instruction par un segment de longueur 1 cm.
2. Léna et Youri souhaitent maintenant obtenir un triangle équilatéral comme motif.

Afin d'obtenir un triangle équilatéral :

- par quelle valeur peut-on remplacer *a* ?
- par quelle valeur peut-on remplacer *b* ?
- par quelle valeur peut-on remplacer *c* ?

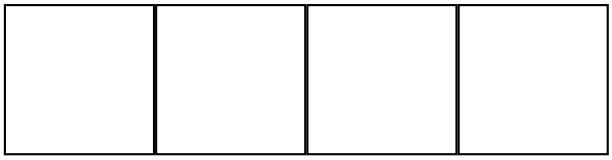


Correction

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/446601390/`

1. (a) La valeur effacée est 60 sinon les carrés seraient jointifs.

   (b)



2.

$a = 3$

$b = 40$ par exemple

$c = 120$.

Analyse

C'est un exercice classique qui reste majoritairement un exercice de mathématiques à part l'utilisation de l'instruction « tourner ».

Il est dommage que l'exercice ne propose pas l'utilisation d'un bloc avec paramètre qui a toute sa place dans cet exercice.

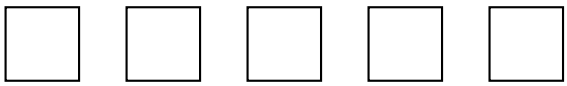
Dans la question 1.(a), on donne aux élèves le choix entre 40 et 60 pas alors que le nombre de pas qui est demandé peut prendre une autre valeur. Le choix entre ces deux valeurs est fait par déduction. On pourrait y ajouter une troisième valeur, par exemple 80 pour pousser la réflexion sur ce point.

Dans la question 2. on n'indique pas la longueur du triangle équilatéral donc remplacer b n'a pas de sens, il peut prendre n'importe quelle valeur. Il faut remarquer aussi que pour la valeur de c, il y a également plusieurs possibilités.

Propositions

Énoncé : En utilisant des blocs avec paramètres

Léna et Youri travaillent sur un programme. Ils ont obtenu le dessin suivant :

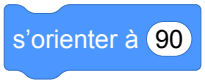


Ils ont ensuite effacé une donnée par erreur dans le script principal.

Voici les copies d'écran de leur travail :

Script principal

Bloc du motif

Pour information, l'instruction 

signifie qu'on se dirige vers la droite.

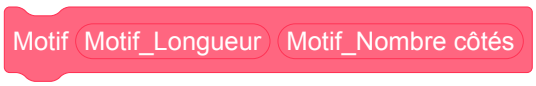
Dans cet exercice, aucune justification n'est demandée.

1. (a) Quelles valeurs peut-on mettre à la valeur manquante dans le script principal à la ligne 9 ? Donner clairement une condition sur cette valeur.

   (b) Dessiner sur la copie ce qu'on aurait obtenu avec la valeur de 60.

   On représentera l'instruction

par un segment de longueur 1 cm.

2. Qu'est-ce que le bloc 

dessine ?

3. Quel autre nom peut-on donner au bloc motif pour que son nom soit plus clair ?
4. Compléter la ligne 2 du bloc du motif pour calculer la valeur de l'angle nécessaire au programme.
5. On souhaite dessiner 7 fois un triangle équilatéral de côté 50 pas. Quelles lignes du script principal et du bloc du motif il faut modifier et par quelles valeurs ?
6. Proposer des changements dans le script principal pour que le résultat de l'exécution de ce programme soit (à première vue) un seul cercle.

Correction : En utilisant des blocs avec paramètres

Fichier Scratch : `sg_amerique-du-sud_2018_novembre_v2.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/543460012/`

1. (a) On peut mettre n'importe quelle valeur plus grand qu'un côté d'une longueur donnée donc une agrandissement du côté du motif, par exemple : Longueur * 2

   (b) On doit dessiner 5 carrés de côté 2 cm avec 1 cm de distance entre chacun.
2. Le bloc dessine des polygones réguliers avec Motif_Nombre côtés côtés de Motif_Longueur pas.
3. On peut le nommer Tracer un polygone régulier.
4. 360 / Motif_Nombre côtés
5. Dans le script principal, il faut modifier
   - la ligne 5 et mettre la variable Longueur en 50 ;
   - la ligne 6 et mettre la variable Nombre côtés en 3 ;
   - la ligne 7 et répéter 7 fois (au lieu de 5).

   Il ne faut rien changer dans le bloc du motif.
6. Dans le script principal, il faut modifier
   - la ligne 5 et mettre la variable Longueur en 1 ;
   - la ligne 6 et mettre la variable Nombre côtés en 360 ;
   - la ligne 7 et répéter 1 fois.

10 Nouvelle Calédonie, décembre 2018, série générale

Références

Série : générale

Académie : Nouvelle Calédonie

Date : décembre 2018

Exercice 8 - 9 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- déplacement

Thèmes informatiques

- variable déclarée
- activité du lutin
- itération bornée

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Nouvelle Calédonie, Décembre 2018

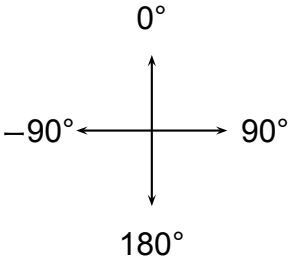
Rappel sur l'orientation du lutin

s'orienter à 90 : pour se déplacer vers la droite

s'orienter à 0 : pour se déplacer vers le haut

s'orienter à -90 : pour se déplacer vers la gauche

s'orienter à 180 : pour se déplacer vers le bas

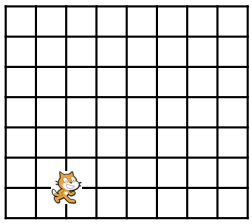


Le chat indique la position de départ.

1.

On exécute le script 1 ci-contre.
Représenter le chemin parcouru par le chat.

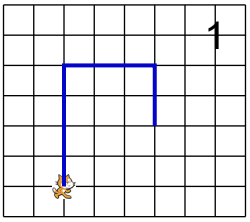
Le côté d'un carreau mesure 20 pas.

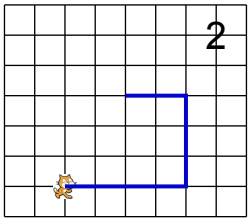


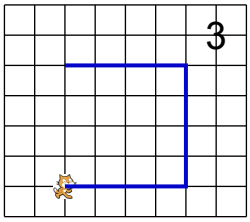
2. (a) Indiquer sur la copie le numéro du dessin correspondant au script 2 ci-dessous.

Script 2

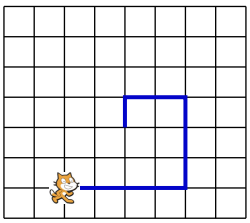
Le côté d'un carreau mesure 20 pas.







(b) On souhaite modifier le script 2 pour parcourir le chemin suivant :

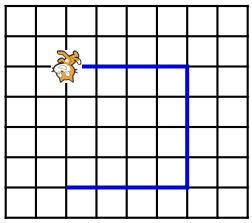


Quelle(s) modification(s) peut-on apporter au script 2 pour parcourir ce chemin ?

Correction

Fichier Scratch : `sg_nouvelle_caledonie_2018_decembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/373386516`

1. Le côté d'un carreau mesure 20 pas.



2. (a) On avance à droite de 4 carreaux, on tourne à gauche et on avance de $80 - 20 = 60$ pas , on tourne à gauche et on avance de $60 - 20 = 40$ pas ; on a obtenu le dessin 2.
   (b) Il suffit de changer le nombre de répétition de la boucle « répéter »et mettre « 3 fois ».

Analyse

L'instruction s'orienter à 90 est toujours rappelée mais avec cette formulation, le lien avec les angles est compliqué à comprendre pour les élèves. De plus il est maladroit. En effet le lutin peut être orienté vers la droite et se déplacer vers la gauche en avançant d'un nombre négatif de pas.
Il suffirait de remplacer *Pour se déplacer ...* par *Pour orienter le lutin ...* .

Le nommage de la variable pas rend peu claire l'instruction avancer de pas pas. On aurait pu choisir un autre nom de variable comme nombre de pas par exemple.
Sur les images présentées dans la deuxième question de ce sujet, la position du chat est sur le point du départ alors que le trait dessiné correspond à ce qui doit être affiché après exécution du programme. Il serait plus clair qu'un autre motif comme l'ombre d'un chat par exemple indique la position de départ. Dans ce cas, il faudrait aussi modifier dans l'énoncé la phrase "Le chat indique la position de départ".

La question 2.(a) pourrait être complétée par une question portant sur l'orientation du chat dans sa position finale.

Le motif de la question 2.(b) mériterait d'être modifié car il donne la réponse à la question 2.(a). Modifier les dimensions de la figure en plus du nombre de tours serait une solution et cela compliquerait un peu la modification.

Il s'agit d'un exercice classique sur les déplacements utilisant les boucles bornées. Il serait intéressant de le compléter en demandant aux élèves d'écrire un script qui suivrait un chemin donné.

Pour aller plus loin, on pourrait proposer un exercice sur la spirale des animaux proposée par Emmanuel V. La spirale des chats présentée ci-dessous permet de travailler les boucles

et

Propositions

Énoncé modifié d'après Nouvelle Calédonie, Décembre 2018

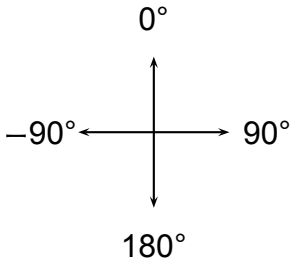
Rappel sur l'orientation du lutin

s'orienter à 90 : s'orienter vers la droite

s'orienter à 0 : s'orienter vers le haut

s'orienter à -90 : s'orienter vers la gauche

s'orienter à 180 : s'orienter vers le bas

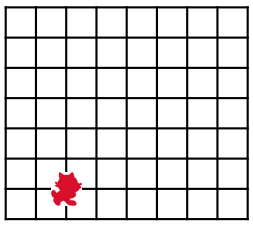


Dans ce qui suit l'ombre du chat indique la position de départ.

1.

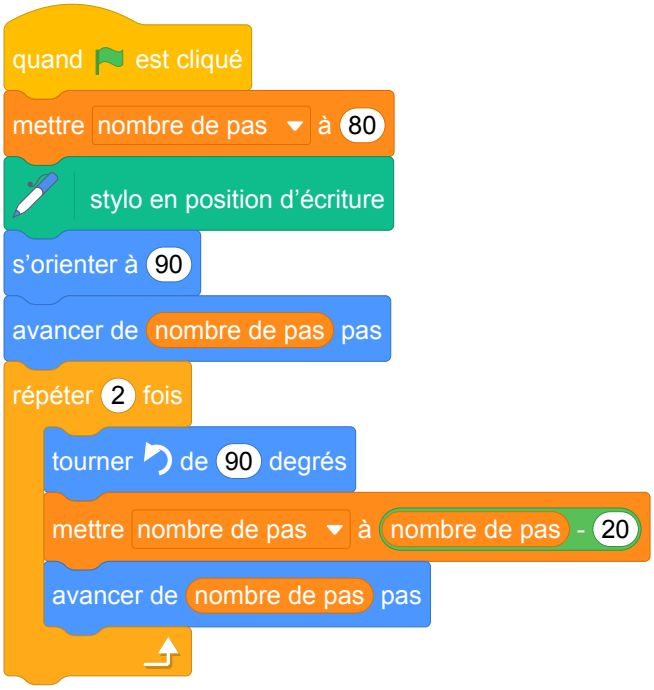
On exécute le script 1 ci-contre.
Représenter le chemin parcouru par le chat.

Le côté d'un carreau mesure 20 pas.

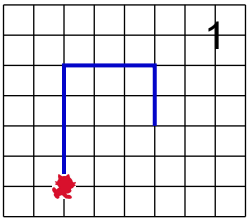


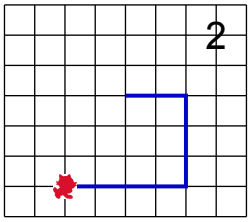
2. (a) Indiquer sur la copie le numéro du dessin correspondant au script 2 ci-dessous.

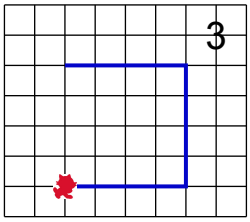
Script 2



Le côté d'un carreau mesure 20 pas.







(b) Parmi les 4 schémas suivants, lequel correspond à la position du chat en fin d'exécution ?

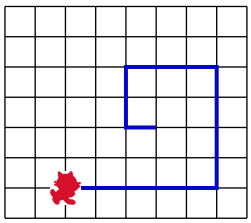
Chat 1

Chat 2

Chat 3

Chat 4

(c) On souhaite modifier le script 2 pour parcourir le chemin suivant :

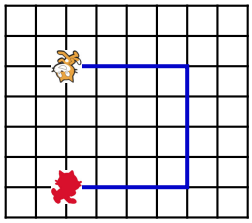


Quelle(s) modification(s) peut-on apporter au script 2 pour parcourir ce chemin ?

Correction de l'énoncé modifié

Fichier Scratch : `sg_nouvelle_caledonie_2018_decembre_modifie.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/540850857`

1. Le côté d'un carreau mesure 20 pas.



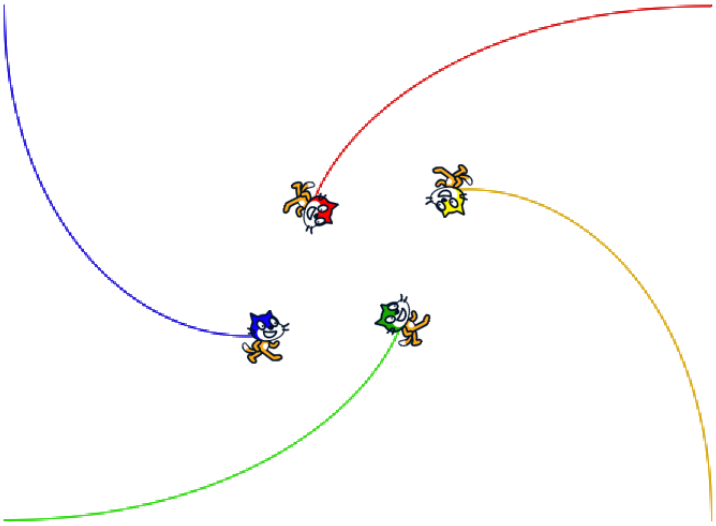
2. (a) On avance à droite de 4 carreaux, on tourne à gauche et on avance de $80 - 20 = 60$ pas , on tourne à gauche et on avance de $60 - 20 = 40$ pas ; on a obtenu le dessin 2.
   (b) Comme le chat a tourné 2 fois à gauche de 90, c'est le chat 3 qui indique sa position en fin d'exécution.
   (c) Il suffit d'initialiser la variable nombre de pas à 100 et de changer le nombre de répétition de la boucle : répéter 4 fois

La spirale des chats

Jusque là les seules spirales rencontrées sont construites avec les blocs "avancer de ... pas" et "tourner de ..." en modifiant le nombre de pas. Il est également possible d'utiliser des courbes de poursuite en s'inspirant de la courbe du chien.

Fichier Scratch : `courbe_du_chien_repeter_jusqu_a.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/578503002`

Un exemple de la spirale des chats est donnée ci dessous.



Fichier Scratch : `spirale_chats_repeter_jusqu_a.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/540559851`

Fichier Scratch : `spirale_3chats_repeter_jusqu_a.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/578501408`

11 Métropole – La Réunion – Antilles – Guyane, juin 2018, série professionnelle

Références

Série : professionnelle

Académie : Métropole – La Réunion – Antilles – Guyane

Date : juin 2018

Exercice 5 - 15 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- déplacement
- grandeurs et mesures

Thèmes informatiques

- scène
- activité du lutin

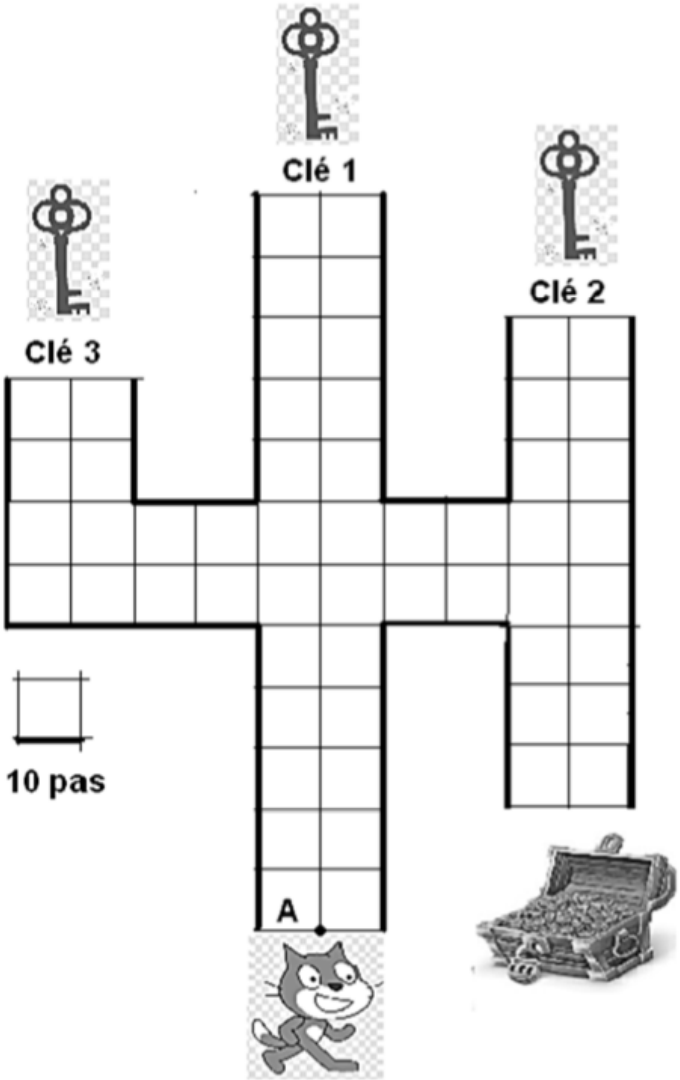
Compétences travaillées

- comprendre un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion – Antilles – Guyane, juin 2018

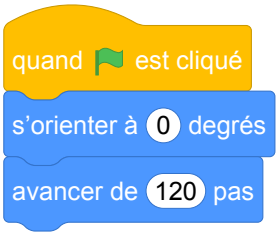
Pour gagner le trésor, le chat doit récupérer les trois clés ouvrant la serrure. Il circule sur la ligne au centre de chaque allée.



1. Pour obtenir la clé 1 en partant du point A, le chat exécute le programme ci-dessous à gauche.

   Expliquer à quoi correspond la valeur 120 dans l'instruction [avancer de 120 pas]

Programme A



Détail de la commande "s'orienter"

[s'orienter à 90 degrés] : à droite

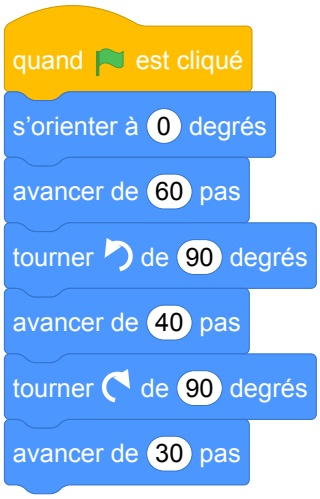
[s'orienter à -90 degrés] : à gauche

[s'orienter à 0 degrés] : vers le haut

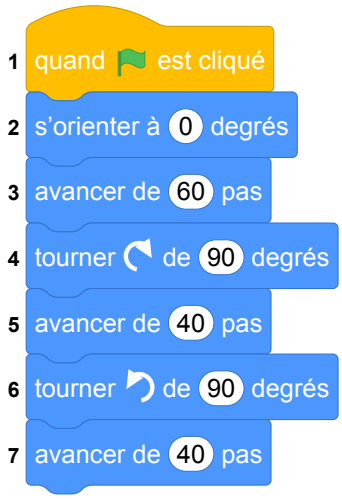
[s'orienter à 180 degrés] : vers le bas

2. Le chat part à nouveau du point A avant d'attraper une nouvelle clé.
   Associer à chaque programme, le numéro de la clé qu'il permet d'obtenir.

Programme B



Programme C



3. Une fois les trois clés obtenues, le chat, revenu au point A, doit partir pour ouvrir le trésor.
   Quelle instruction du **programme C** est à modifier pour atteindre le trésor ?
   Préciser le changement à effectuer.

Correction

Fichier Scratch : `pro_metropole_2018_juin.sb3` Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/772435785/`

1. 120 indique le nombre de pas que doit parcourir le chat. Étant donné que le chat doit parcourir 12 carreaux de 10 pas de côté, le chat doit bien parcourir 120 pas.
2. Le programme B permet d'obtenir la clé 3.
   Le programme C permet d'obtenir la clé 2.
3. Il suffit de remplacer l'instruction 6 par l'instruction

Analyse

- La question 1 a-t-elle encore de l'intérêt alors que dans Scratch 3 a été ajouté le mot "pas" à l'instruction

?
- La position de départ est le point A. Or on note que le chat ne se trouve pas en A au départ.
- Dans le même ordre d'idées, les points d'arrivée ne sont pas réellement précisés.
- Le chat tel qu'il est représenté dans sa situation initiale n'est pas orienté comme l'indique les programmes. Cela pourrait créer une difficulté pour les élèves entre un déplacement vertical et horizontal.
- Dans la consigne, la phrase *"Il circule sur la ligne au centre de chaque allée."* manque de clarté.
- Il manque à l'élève un moyen pour désigner dans la question 3 l'instruction à modifier.

Cet exercice présente un intérêt informatique pauvre.

Propositions

Voir les suggestions de l'IREM Marseille sur le labyrinthe d'OMER.

Source groupe *Enseignement de l'informatique* IREM de Marseille : `https://ens-info.irem.univ-mrs.fr/wp-content/uploads/sujetProjuin2018v1.1.pdf`

Fichier Scratch : `pro_polynesie_2018_juin_creation_labyrinthe_Homer.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/690064341/`

Titre : Le labyrinthe d'Homer

Niveau :

- cycle 4
- CAP

Langage de programmation : Scratch

Passation de l'activité : Quelles consignes ? Travail en groupe ? Etc.

- Travail individuel ou en groupe.
- Possibilité d'utiliser le fichier Scratch en support.

Savoir visé

- Comprendre l'effet d'une translation, d'une symétrie (axiale et centrale), d'une rotation, d'une homothétie sur une figure.
- Décomposer un problème en sous-problèmes afin de structurer un programme ; reconnaître des schémas.
- Écrire, mettre au point (tester, corriger) et exécuter un programme en réponse à un problème donné.
- Écrire un programme dans lequel des actions sont déclenchées par des événements extérieurs.

Difficultés des élèves Mettre en corrélation le progamme Scratch et la scène proposée.

Type de tâche

- lire et comprendre un algorithme
- exécuter un algorithme
- calculer une 4ème proportionnelle

Techniques

a priori

- identifier les instructions d'un algorithme (i.e. reconnaître les blocs Scratch)
- calculer une 4ème proportionnelle

a posteriori Restitution d'élève modifier un paramètre de boucle

Éléments théoriques

- Séquence d'instruction
- Evénement
- Boucle
- Condition

Objectifs par question :

1. Question 1 :
   Objectif :
   - appropriation de la situation

   Attendus :
   - calcul de proportionnalité
2. Question 2 :
   Objectif :
   - appropriation de la situation

   Attendus :
   - identification des paramètres : déplacement, nombre de boucles
   - calcul de proportionnalité
3. Question 3 :
   Objectif :
   - élaborer un programme

   Attendus :
   - identification des paramètres : sens de rotation
   - calcul de proportionnalité
4. Question 4 : Objectif :
   - élaborer un programme

   Attendus :
   - compréhension des différentes structures conditionnelles
5. Question 5 : (dépassement des attentes)
   Objectif :
   - élaborer un programme

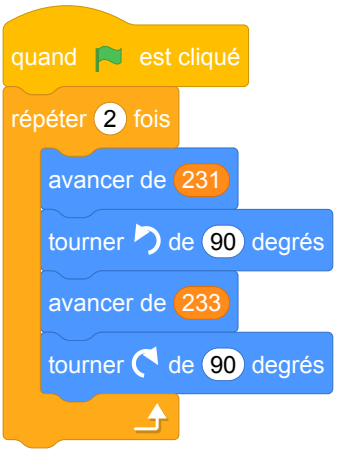
   Attendus :
   - identification de l'état final du programme précédent
   - identification des paramètres : sens de rotation
   - ordre des instructions

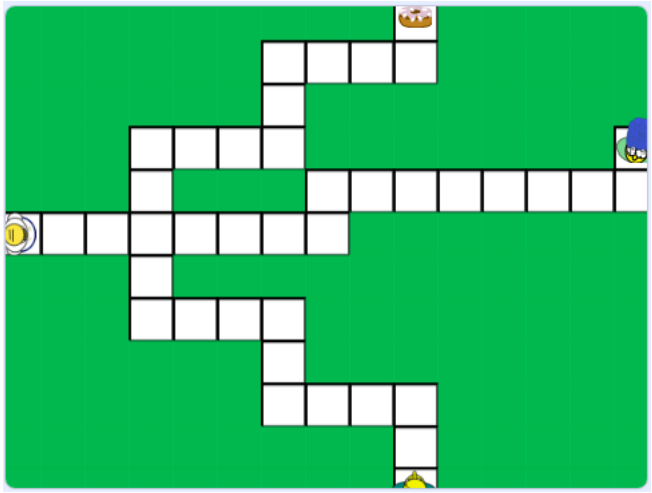
DOCUMENT ÉLÈVE

Situation : un programme permet de diriger Homer dans un labyrinthe, aidons-le à retrouver son donut ou l'horrible Mr Burns.

Homer : Marge : Mr Burns : le donut :

Le programme ci-dessous lui permet à partir de son point de départ, de rejoindre Marge.





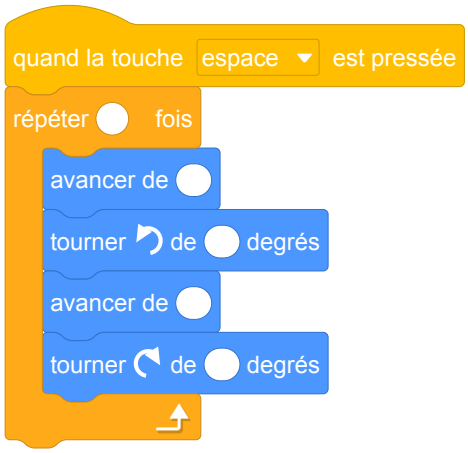
Le Labyrinthe

Question1

Combien de pas représente un carreau.

Question 2

Modifier les valeurs numériques dans ce programme afin de pouvoir rejoindre le donut en partant du point de départ.

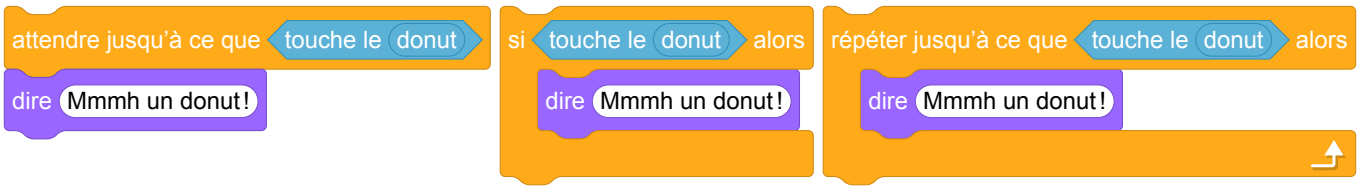


Question 3

Quelle instruction faut-il modifier dans le programme précédent afin qu'il puisse rejoindre Mr Burns en partant du point de départ.

Question 4

A la fin du programme on veut qu'Homer dise « Mmmh un donut » lorsqu'il a retrouvé son donut. Quelle instruction parmis ces trois doit-on ajouter ?



Question 5

Proposer un programme permettant à Homer de revenir au point de départ après avoir rejoint Mr Burns.

12 Polynésie, juin 2018, série professionnelle

Références

Série : professionnelle

Académie : Polynésie

Date : juin 2018

Exercice 6 - 20 points / 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- agrandissement réduction
- caractérisation des figures
- spirale
- déplacement
- nombres et calculs
- calcul sans programme
- grandeurs et mesures

Thèmes informatiques

- variable déclarée
- activité du lutin
- calcul
- aléatoire
- branchement conditionnel
- itération conditionnelle
- capteurs

Compétences travaillées

- comprendre un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	1
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, Juin 2018 série Professionnelle

Dans cet exercice, le symbole F représente l'unité franc CFP.

1. Afin d'étudier les fonds marins d'un atoll, on programme une sonde aquatique qui se déplace dans le lagon. Trois programmes sont possibles.

Programme 1

```
quand est cliqué
aller à x : 0 y : 0
répéter indéfiniment
    avancer de 10 pas
    si touche le bord alors
        tourner de de nombre aléatoire entre 45 et 180 degrés
        avancer de 10 pas
```

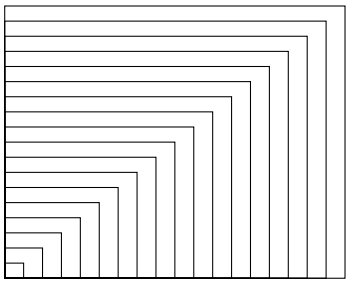
Programme 2

```
quand est cliqué
aller à x : -220 y : -160
mettre L à 0
mettre l à 0
stylo en position d'écriture
répéter jusqu'à ce que touche le bord
    ajouter 25 à L
    ajouter 20 à l
    avancer de L pas
    tourner de de 90 degrés
    avancer de l pas
    tourner de de 90 degrés
    avancer de L pas
    tourner de de 90 degrés
    avancer de l pas
    tourner de de 90 degrés
```

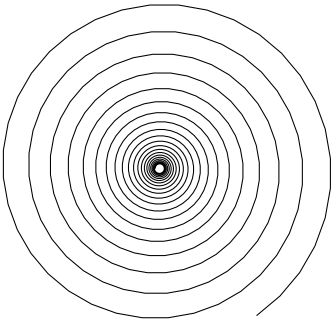
Programme 3

```
quand est cliqué
aller à x : 0 y : 0
s'orienter à 90
effacer tout
stylo en position d'écriture
mettre pas à 1
répéter jusqu'à ce que touche le bord
    avancer de pas pas
    tourner de de 10 degrés
    mettre pas à pas * 1.005
```

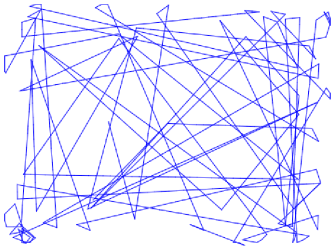
Les 3 schémas suivants montrent les déplacements de la sonde en fonction du programme.



Déplacement A



Déplacement B



Déplacement C

Attribuer à chaque programme (1, 2 ou 3) le déplacement de la sonde qui lui correspond (A, B ou C).

2. L'utilisation de cette sonde a un coût. En effet la mise en place du matériel coûte 80 000 F puis l'exploration marine coûte 15 000 F par heure d'utilisation.

   On ajoute, aux programmes précédents, des instructions permettant de calculer le coût à payer en fonction du temps d'exploration.

   (a) Pour 4h d'exploration, le programme nous donne un coût de 140 000 F. Vérifier la valeur de ce coût.

   (b) Recopier et compléter l'instruction suivante, permettant de calculer le coût à payer en fonction du temps.

3. Un club de plongée propose la même exploration du lagon pour 180 000 F la journée. Déterminer la durée à partir de laquelle l'utilisation de la sonde revient plus cher que la prestation club de plongée.

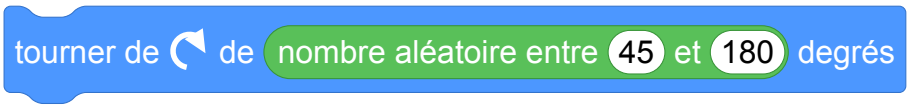
Correction

Fichier Scratch : `pro_polynesie_2018_juin.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/481102190/`

1. Seul le programme 2 permet de tracer des rectangles (séquence d'instructions se trouvant dans la boucle), il correspond donc au déplacement A.

   Le programme 1 contient une rotation aléatoire à chaque itération indiquant que le déplacement correspondant est le C :

   Le déplacement B est produit par le programme 3. La rotation régulère est obtenue par le bloc :

2. (a) $80000 + 15000 \times 4 = 80000 + 60000 = 140000$

   Le coût pour 4h est bien de 140 000 F.

   (b)

3. Notons n le nombre d'heures à partir de laquelle l'utilisation de la sonde revient plus cher que la prestation club de plongée.

   Considérant que l'exploration dure moins d'une journée, il s'agit de résoudre l'inéquation
   $80000 + n \times 15000 > 180000$
   $n \times 15000 > 180000 - 80000$
   $n \times 15000 > 100000$
   $n > 100000 \div 15000$
   $n > 100 \div 15$
   $n > 20 \div 3$
   $n > 6,666...$

   Autrement dit, à partir de 7h, l'utilisation de la sonde coûte plus cher que la prestation club de plongée.

   Comme le tarif quotidien de 180 000 F correspond à $180000 \div 15000 = 12$ h d'utilisation de la sonde, on peut conclure que pour 6h ou moins, c'est l'utilisation de la sonde qui est plus intéressante et qu'à partir de 7h c'est la prestation club de plongée qui est plus intéressante.

Analyse

— Les programmes sont intéressants mais le lien entre les déplacements et les programmes sont "évidents", et on trouve rapidement le résultat par élimination.

— L'intérêt des questions 2 et 3 en informatique est limité.

— Question 2b) Ici on ne devrait pas parler d'instruction mais d'expression car on ne peut pas mettre ce bloc directement dans le programme mais à l'intérieur d'une instruction.

— Il n'y a aucune précision sur l'unité du temps.

Propositions

Dans cette version des programmes, nous avons modifié le texte original en ajoutant dans les programmes 1 et 3 le bloc :

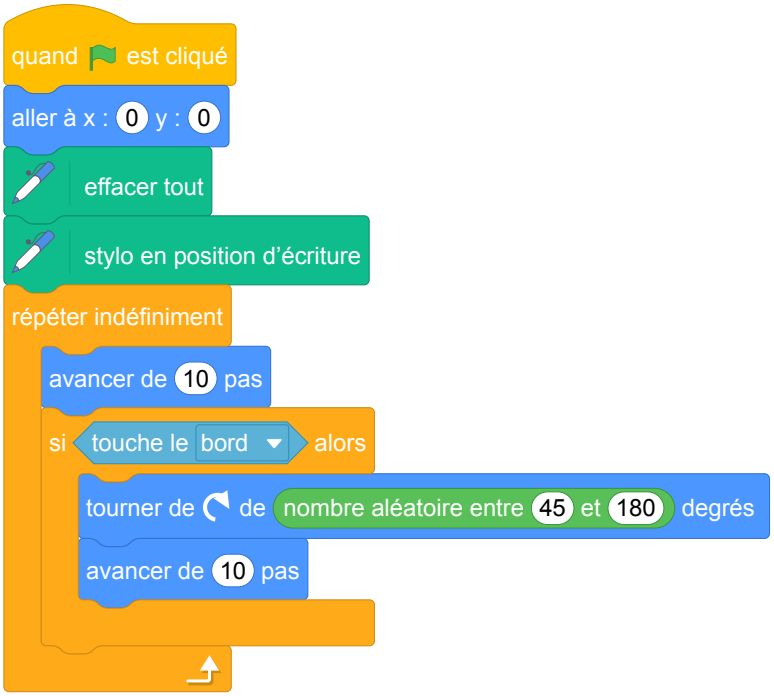
Ceci permet de démarer le tracé avec une scène vierge.

Début de l'énoncé modifié

Dans cet exercice, le symbole F représente l'unité franc CFP.

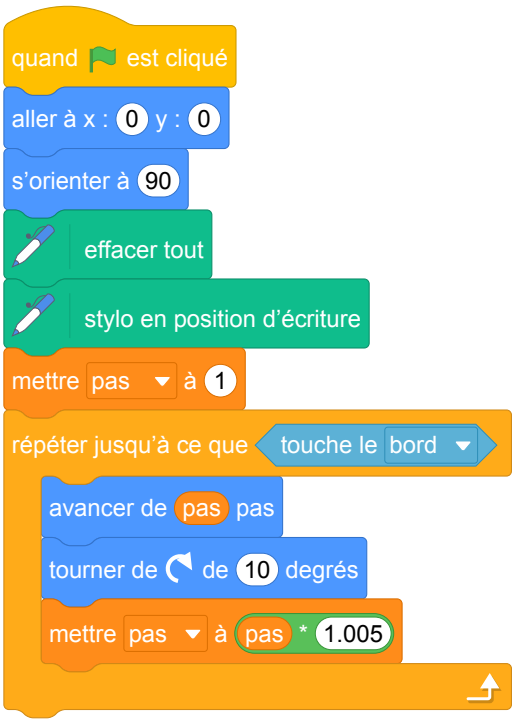
1. Afin d'étudier les fonds marins d'un atoll, on programme une sonde aquatique qui se déplace dans le lagon. Trois programmes sont possibles.

Programme 1

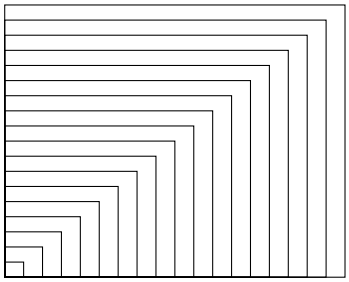


Programme 2

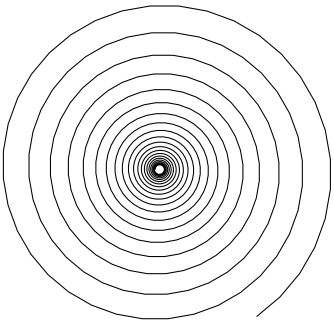
Programme 3



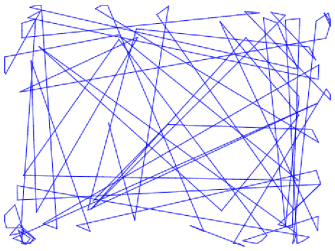
Les 3 schémas suivants montrent les déplacements de la sonde en fonction du programme.



Déplacement A



Déplacement B



Déplacement C

Attribuer à chaque programme (1, 2 ou 3) le déplacement de la sonde qui lui correspond (A, B ou C).

2. L'utilisation de cette sonde a un coût. En effet la mise en place du matériel coûte 80 000 F puis l'exploration marine coûte 15 000 F par heure d'utilisation.

   On ajoute, aux programmes précédents, des instructions permettant de calculer le coût à payer en fonction du temps d'exploration.

   (a) Pour 4h d'exploration, le programme nous donne un coût de 140 000 F. Vérifier la valeur de ce coût.

   (b) Recopier et compléter l'instruction suivante, permettant de calculer le coût à payer en fonction du temps.

3. Un club de plongée propose la même exploration du lagon pour 180 000 F la journée. Déterminer la durée à partir de laquelle l'utilisation de la sonde revient plus cher que la prestation club de plongée.

13 Polynésie, septembre 2018, série professionnelle

Références

Série : professionnelle

Académie : Polynésie

Date : septembre 2018

Exercice 4 - 12 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

Thèmes informatiques

- variable déclarée
- branchement conditionnel
- évenèment avec message
- capteurs

Compétences travaillées

- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	1
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	0
itération conditionnelle	0
imbrication de structures	1
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, Septembre 2018

Il est possible de piloter la climatisation de son domicile à distance à l'aide de deux programmes.

1. Le programme A est le suivant :

```
quand je reçois Allumer la climatisation
si <température > 25> alors
    mettre climatisation à 1
sinon
    mettre climatisation à 0
```

   Indiquer ce qu'il se passe si la commande *Allumer la climatisation* est sélectionnée et que la température de la pièce est de 27°.

2. Le programme B permet de régler la puissance de la climatisation en fonction de la température. Ainsi lorsque la température est :
   - supérieure à 28°C la climatisation est sur le niveau 2 ;
   - entre 28°C et 25°C, la climatisation est sur le niveau 1 ;
   - inférieure à 25°C, elle s'arrête.

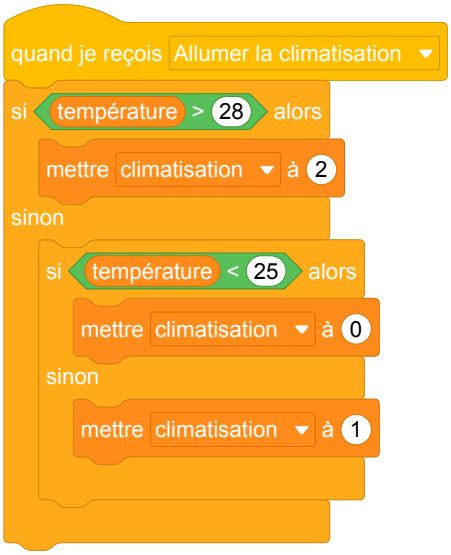
   Compléter les cases contenant des pointillés du programme B :

```
quand je reçois Allumer la climatisation
si <température > .....> alors
    mettre climatisation à .....
sinon
    si <température < .....> alors
        mettre climatisation à .....
    sinon
        mettre climatisation à .....
```

Correction

Fichier scratch : `pro_polynesie_2018_septembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/772453242`

1. Quand la commande Allumer la climatisation est sélectionnée, l'instruction "si" est exécutée, la condition "température > 25" est évaluée, sa valeur est vraie, la variable climatisation est donc affectée à la valeur 1.
2. Voici le programme B complété :



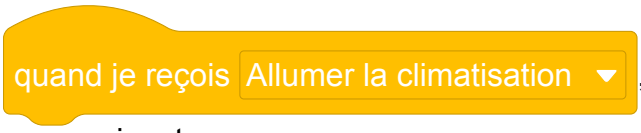
Analyse

— Dans la question 1, l'information *"la commande *Allumer la climatisation* est sélectionnée"* n'est pas très claire, en particulier le mot "sélectionnée" est ambigu, il pourrait désigner le fait de marquer cette commande en vue de lui faire subir une opération. Il pourrait être remplacé par "activée".

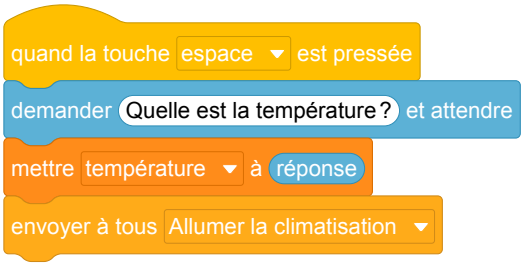
— Il serait bien de différencier par deux variables l'état de la climatisation Allumé/Eteint et le niveau de la climatisation.

— On note l'utilisation d'instructions "si...alors...sinon" imbriquées qui est très rare dans les sujets de brevet.

Propositions

— On pourrait préciser l'origine de l'instruction 

, ce qu'il se passe avant avec par exemple le programme suivant :



Cela permet de tester le programme dans Scratch. Nous avons donc ajouté ce script dans le fichier Scratch de correction.

— Cependant, cet exercice serait plutôt à travailler en classe avec une mise en application par l'utilisation d'un matériel adéquat (robot Mbot ou carte Arduino avec capteur de température, afficheur ...). Dans ce cas le script précédent n'a plus d'utilité.

Brevets 2019

1 Polynésie, juillet 2019, série générale

Références

Série : générale

Académie : Polynésie

Date : juillet 2019

Exercice 2 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable réservée
- variable déclarée
- calcul
- évènement avec message

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- utiliser une variable avec auto référence

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

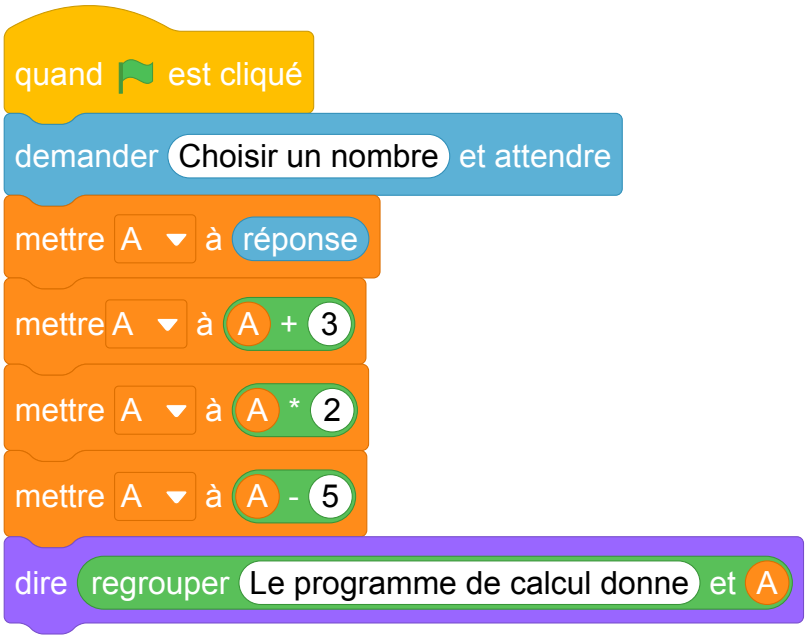
Énoncé - Polynésie, juillet 2019

1. On a utilisé une feuille de calcul pour obtenir les images de différentes valeurs de x par une fonction affine f.
   Voici une copie de l'écran obtenu :

B2		=3*B1−4						
	A	B	C	D	E	F	G	H
1	x	−2	−1	0	1	2	3	4
2	$f(x)$	−10	−7	−4	−1	2	5	8

   (a) Quelle est l'image de -1 par la fonction f ?
   (b) Quel est l'antécédent de 5 par la fonction f ?
   (c) Donner l'expression de $f(x)$.
   (d) Calculer $f(10)$.

2. On donne le programme suivant qui traduit un programme de calcul.



   (a) Écrire sur votre copie les deux dernières étapes du programme de calcul :

   - Choisir un nombre.
   - Ajouter 3 à ce nombre.
   - …
   - …

   (b) Si on choisit le nombre 8 au départ, quel sera le résultat ?
   (c) Si on choisit x comme nombre de départ, montrer que le résultat obtenu avec ce programme de calcul sera $2x + 1$.
   (d) Quel nombre doit-on choisir au départ pour obtenir 6 ?

3. Quel nombre faudrait-il choisir pour que la fonction f et le programme de calcul donnent le même résultat ?

Correction

Fichier Scratch : `sg_polynesie_2019_juillet.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/453265407`

1. (a) Les antécédents sont dans la ligne 1, les images dans la ligne 2.
   L'image de -1 par la fonction f est $f(-1) = -7$.
   (b) L'antécédent de 5 par la fonction f est 3.
   (c) On a $f(x) = 3x - 4$.
   (d) Donc $f(10) = 3 \times 10 - 4 = 30 - 4 = 26$.
2. (a) Écrire sur votre copie les deux dernières étapes du programme de calcul :

   - Choisir un nombre.
   - Ajouter 3 à ce nombre.
   - Multiplier le résultat par 2
   - Soustraire 5 au résultat

   (b) 8 donne successivement $8 \to 11 \to 22 \to 17$.
   (c) x donne successivement $x \to x+3 \to 2(x+3) \to 2(x+3) - 5$.
   Or $2(x+3) - 5 = 2x + 6 - 5 = 2x + 1$.
   (d) • Il faut trouver x tel que $2(x+3) - 5 = 2x + 6 - 5 = 2x + 1 = 6$ soit $2x = 5$ et enfin $x = 2,5$.
   • On peut « remonter » les opérations :
   $5,5 - 3 = 2,5 \leftarrow \dfrac{11}{2} = 5,5 \leftarrow 6 + 5 = 11 \leftarrow 6$.
3. Il faut trouver x tel que :
   $3x - 4 = 2x + 1$ soit en ajoutant $-2x$ à chaque membre : $x - 4 = 1$ et en ajoutant 4 à chaque membre : $x = 5$.
   Par f et par le programme de calcul 5 donne 11.

Analyse

Cet exercice a un intérêt certain dans un travail sur l'algèbre par les programmes de calculs au sens de la didactique de l'algèbre.

D'un point de vue informatique, il n'y a aucune structure de contrôle et le passage entre la variable informatique A et la variable mathématique x peut amener des confusions. Celles-ci sont notamment liées à l'appel par autoréférence de la variable A dans les calculs.

Les limites de l'utilisation de Scratch empêchent de travailler cet exercice avec une fonction qui prendrait pour paramètre la variable A.

L'utilisation des blocs demander et dire qui correspondent aux instructions "input" et "print" dans d'autres langages peuvent induire des habitudes de programmation qui ont leur limite. Dans Scratch, il n'est pas indispensable d'utiliser ces blocs puisqu'en faisant afficher la variable A sur la scène, on peut manuellement lui affecter une valeur qui reste alors visible, et utiliser cette valeur dans le programme.

Enfin les questions posées n'interrogent que des notions d'algèbre et pas d'informatique.

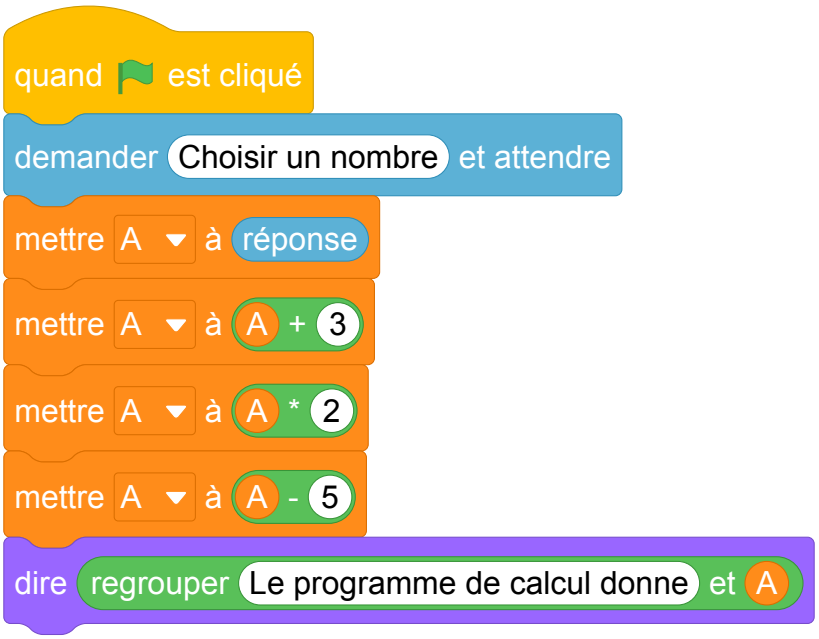
La question de la variable informatique dans cet exercice semble centrale et pourrait être travaillée, par une activité de construction d'un tableau d'exécution faisant apparaître l'évolution du contenu de la variable A.

De plus, on pourrait aussi imaginer un travail sur les priorités de calculs en faisant le lien avec une représentation arborescente des expressions, et montrer que pour évaluer une expression, on parcourt l'arbre des feuilles vers la racine, alors que pour nommer le calcul, il faut faire l'inverse.

Propositions

Énoncé modifié à partir de la question 2

On donne le programme suivant :



1. Écrire sur votre copie les deux dernières étapes du programme de calcul :

• Choisir un nombre. • Ajouter 3 à ce nombre. • … • …

2. Compléter le tableau suivant en indiquant pour chaque ligne d'instruction, quelle est la valeur contenue dans la variable *A* à la fin de son exécution si le nombre choisi par l'utilisateur du programme est 8.

Instruction	Quelle valeur est contenue dans la variable A ?
quand est cliqué	
demander Choisir un nombre et attendre	
mettre A à réponse	
mettre A à A + 3	
mettre A à A * 2	
mettre A à A - 5	
dire regrouper Le programme de calcul donne et A	

3. On note x le nombre choisi par l'utilisateur. Dans le tableau suivant, pour chaque étape, écrire l'expression littérale en fonction de x correspondante au contenu de la variable A en fin d'instruction. L'expression devra être donnée sous forme développée réduite.

Instruction	L'expression du contenu de A en fonction de x
quand est cliqué	
demander Choisir un nombre et attendre	
mettre A à réponse	
mettre A à A + 3	
mettre A à A * 2	
mettre A à A - 5	
dire regrouper Le programme de calcul donne et A	

Idée : l'utilisation d'une ardoise sur laquelle on note le contenu de la variable A, et qu'on efface pour modifier son contenu peut aider les élèves à donner du sens à cette notion.

Corrigé de l'énoncé modifié

1. Écrire sur votre copie les deux dernières étapes du programme de calcul :

- Choisir un nombre.
- Ajouter 3 à ce nombre.
- Multiplier le résultat par 2
- Soustraire 5 au résultat

2. Si le nombre choisi par l'utilisateur du programme est 8.

Instruction	Quelle valeur est contenue dans la variable A ?
quand est cliqué	
demander Choisir un nombre et attendre	
mettre A à réponse	8
mettre A à A + 3	$8 + 3 = 11$
mettre A à A * 2	$11 \times 2 = 22$
mettre A à A - 5	$22 - 5 = 17$
dire regrouper Le programme de calcul donne et A	17

3. Si on note x, le nombre choisi par l'utilisateur,

Instruction	L'expression du contenu de A en fonction de x
quand est cliqué	
demander Choisir un nombre et attendre	
mettre A à réponse	x
mettre A à A + 3	$x + 3$
mettre A à A * 2	$(x + 3) \times 2 = 2x + 6$
mettre A à A - 5	$(2x + 6) - 5 = 2x + 1$
dire regrouper Le programme de calcul donne et A	$2x + 1$

Le contenu de la variable A à l'issue de l'exécution du programme est donc $2x + 1$.

2 Polynésie, septembre 2019, série générale

Références

Série : générale

Académie : Polynésie

Date : septembre 2019

Exercice 7 - 14 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- programme de calcul

Thèmes informatiques

- variable réservée
- variable déclarée
- calcul
- capteurs

Compétences travaillées

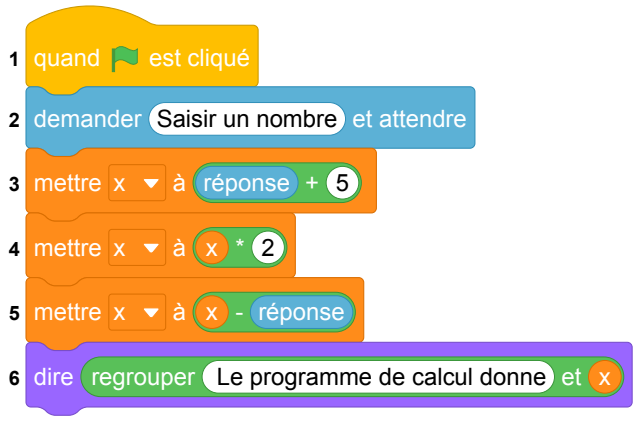
- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

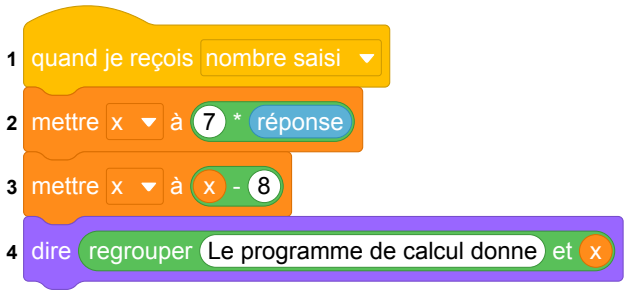
Énoncé - Polynésie, septembre 2019

On donne le programme ci-dessous où on considère 2 lutins. Pour chaque lutin, on a écrit un script correspondant à un programme de calcul différent.

Lutin n° 1



Lutin n° 2



1. Vérifier que si on saisit 7 comme nombre, le lutin n° 1 affiche comme résultat 17 et le lutin n° 2 affiche 41.
2. Quel résultat affiche le lutin n° 2 si on saisit le nombre -4 ?
3. (a) Si on appelle x le nombre saisi, écrire en fonction de x les expressions qui traduisent le programme de calcul du lutin n° 1, à chaque étape (instructions 3 à 5).

   (b) Montrer que cette expression peut s'écrire $x + 10$.
4. Célia affirme que plusieurs instructions dans le script du lutin n° 1 peuvent être supprimées et remplacées par celle ci-dessous.

   Indiquer, sur la copie, les numéros des instructions qui sont alors inutiles.
5. Paul a saisi un nombre pour lequel les lutins n° 1 et n° 2 affichent le même résultat. Quel est ce nombre ?

Correction

Fichier Scratch : `sg_polynesie_2019_septembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/481401018/`

1. Le premier programme donne : $7 \to 12 \to 24 \to 17$.
   Le deuxième programme donne : $7 \to 49 \to 41$.
2. On obtient successivement : $-4 \to -28 \to -36$.
3. (a) Le programme 1 donne : $x \to x+5 \to 2(x+5) \to 2(x+5)-x$.
   (b) Le résultat final précédent d'écrit :
   $2(x+5)-x = 2x+10-x = x+10$.
4. On peut supprimer les instructions 3, 4 et 5.
5. Le deuxième programme donne si on introduit le nombre x, $7x-8$.
   Donc les deux programmes donnent le même résultat si :
   $x+10 = 7x-8$, soit $18 = 6x$ ou $6 \times 3 = 6 \times x$, d'où finalement $x = 3$.
   Vérification : le lutin n° 1 donne $3+10 = 13$ et le lutin n° 2 donne $7 \times 3 - 8 = 21 - 8 = 13$.

3 Métropole La Réunion - Antilles-Guyane, septembre 2019, série générale

Références

Série : générale

Académie : Métropole, La Réunion, Antilles-Guyane

Date : septembre 2019

Exercice 6 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- probabilités
- simulations

Thèmes informatiques

- variable déclarée
- aléatoire
- branchement conditionnel
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- compléter un algorithme
- écrire un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	1
chgt avec auto-référence	1
calcul	0
aléatoire	1
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	1
itération conditionnelle	0
imbrication de structures	1
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

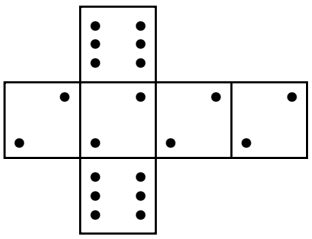
Énoncé - Métropole, La Réunion, Antilles-Guyane, septembre 2019

Deux amis Armelle et Basile jouent aux dés en utilisant des dés bien équilibrés mais dont les faces ont été modifiées. Armelle joue avec le dé A et Basile joue avec le dé B.

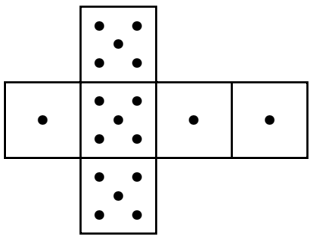
Lors d'une partie, chaque joueur lance son dé et celui qui obtient le plus grand numéro gagne un point.

Voici les patrons des deux dés :

Patron du dé A



Patron du dé B



1. Une partie peut-elle aboutir à un match nul ?
2. (a) Si le résultat obtenu avec le dé A est 2, quelle est la probabilité que Basile gagne un point ?
   (b) Si le résultat obtenu avec le dé B est 1, quelle est la probabilité qu'Armelle gagne un point ?

Les joueurs souhaitent comparer leur chance de gagner. Ils décident de simuler un match de soixante mille duels à l'aide d'un programme informatique.
Voici une partie du programme qu'ils ont réalisé.

Programme principal

```
quand [drapeau] est cliqué
mettre Victoire de A à 0
mettre Victoire de B à 0
répéter 60000 fois
    Lancer le dé A
    Lancer le dé B
    si ( ) < ( ) alors
        ajouter 1 à Victoire de A
    sinon
        ajouter 1 à Victoire de B
```

Sous-programmes

```
définir Lancer le dé A
mettre tirage de dé à nombre aléatoire entre 1 et 6
si tirage de dé < 5 alors
    mettre FaceA à 2
sinon
    mettre FaceA à 6

définir Lancer le dé B
```

On précise que l'expression nombre aléatoire entre 1 et 6 renvoie de manière équiprobable un nombre pouvant être 1 ; 2 ; 3 ; 4 ; 5 ou 6.

Les variables FaceA et FaceB enregistrent les résultats des dés A et B. Par exemple, la variable FaceA peut prendre soit la valeur 2 soit la valeur 6, puisque ce sont les seuls nombres présents sur le dé A.
Les variables Victoire de A et Victoire de B comptent les victoires des joueurs.

3. (a) Lorsqu'on exécute le sous-programme Lancer le dé A, quelle est la probabilité que la variable FaceA prenne la valeur 2 ?
   (b) Recopier la ligne 7 du programme principal en la complétant.
   (c) Rédiger un sous-programme Lancer le dé B qui simule le lancer du dé B et enregistre le nombre obtenu dans la variable FaceB.
4. Après exécution du programme principal, on obtient les résultats suivants :
   Victoire de A = 39 901 Victoire de B = 20 099
   (a) Calculer la fréquence de gain du joueur A, exprimée en pourcentage. On donnera une valeur approchée à 1 % près.
   (b) Conjecturer la probabilité que A gagne contre B.

Correction

Fichier Scratch : `sg_metropole_2019_septembre.sb3`
Fichier Scratch : `sg_metropole_2019_septembre_corrige.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/453274457`
Lien vers le fichier Scratch du corrigé sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/508137895`

1. Armelle peut tirer 2 ou 6 et Basile 1 ou 5 : il ne peut y avoir égalité, donc de match nul.
2. (a) Basile a $\frac{3}{6} = \frac{1}{2}$ chances de sortir un 5 donc de battre Armelle.
   (b) Dans tous les cas Armelle tire un 2 un 6 et tous les deux sont supérieurs à 1 : sa probabilité de battre Basile est donc égale à 1.
3. (a) Il y a $\frac{4}{6} = \frac{2}{3}$ de chances que le nombre tiré soit inférieur à 5 donc que FaceA prenne la valeur 2.
   (b)

```
si <FaceB < FaceA> alors
    ajouter 1 à Victoire de A
sinon
    ajouter 1 à Victoire de B
```

   (c) C'est le même sous-programme que Lancer le dé A en remplaçant Lancer le dé A par Lancer le dé B, 5 par 4 (troisième ligne), 2 par 1 (quatrième ligne) et 6 par 5 (cinquième ligne).

```
définir Lancer le dé B
mettre tirage de dé à nombre aléatoire entre 1 et 6
si <tirage de dé < 4> alors
    mettre FaceB à 1
sinon
    mettre FaceB à 5
```

   Il y a une autre solution puisqu'il y a une chance sur 2 de tomber sur 1 et une chance sur 2 de tomber sur 5.

```
définir Lancer le dé B
mettre tirage de dé à nombre aléatoire entre 1 et 2
si tirage de dé = 1 alors
    mettre FaceB à 1
sinon
    mettre FaceB à 5
```

4. (a) La fréquence de victoires de A est égale à $\dfrac{39\,901}{39\,901+20\,099}=\dfrac{39\,901}{60\,000}\approx 0,665$, soit 66,5 % à 0,1 près.

   (b) La probabilité que A gagne contre B est d'environ 66,666 % soit 2 chances sur 3.

Analyse

Ce sujet est intéressant. Il doit cependant être compliqué pour les élèves de faire le lien entre l'usage des structures de contrôle *si ... alors* et les faces des dés.

Les noms de variables pouraient être plus clairs : FaceA serait avantageusement renommé Resultat du dé A.

On note la multiplicité des rôles des nombres entiers dans ce programme :

— il y a les entiers qui permettent de contrôler le tirage aléatoire ;

— il y a les entiers qui représentent le score du lancer (face du dé).

Le lien à faire entre les deux est probablement difficile pour les élèves.

Parmi les difficultés informatiques on note que l'absence de fonctions en Scratch. Cela impose l'usage de blocs sans paramètre agissant par effets de bords sur des variables globales.

Les questions 1 et 2 portent uniquement sur les probabilités, et la question 4 sur des notions de statistiques (fréquence, arrondi, etc.). Seule la question 3 traite d'informatique.

Si la question 3.(a) semble porter sur la compréhension du sous-programme Lancer de A, celle-ci est biaisée par la représentation du dé qui permet de calculer la probabilité demandée sans même lire le programme.

La question 3b consiste à compléter la condition d'attribution des points et nécessite d'avoir compris le fonctionnement du programme. Quant à la question suivante qui consiste à écrire le bloc Lancer de B par analogie, elle nécessite une très bonne compréhension du modèle

Propositions

Il pourrait être intéressant d'aller plus loin dans l'usage des concepts informatiques en travaillant avec des listes pour mémoriser les lancers.

On pourrait également exploiter d'avantage l'aspect statistiques en faisant programmer une expérience qui fait varier le nombre lancers afin de mettre en évidence la convergence de la suite.

D'autres dés à 6 faces non classiques peuvent faire l'objets d'exercices intéressants comme les dés de Sicherman par exemple. Un exemple d'activité sur carte microbit (ou avec le simulateur de la carte) est proposé dans la brochure de la C2I Tices suivante : `https://www.univ-irem.fr/IMG/pdf/microbit-tomcac1.pdf` page 16.

4 Nouvelle Calédonie, mars 2019, série générale

Références

Série : générale

Académie : Nouvelle Calédonie

Date : mars 2019

Exercice 6 - 10 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- déplacement
- grandeurs et mesures

Thèmes informatiques

- scène
- variable réservee
- variable déclarée
- activité du lutin

Compétences travaillées

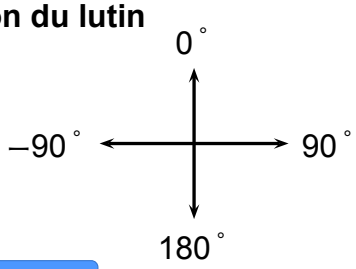
- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Nouvelle Calédonie, Mars 2019

Rappels Scratch

Orientation du lutin



s'orienter à 90 : pour se déplacer vers la droite

s'orienter à 90 : pour se déplacer vers le haut

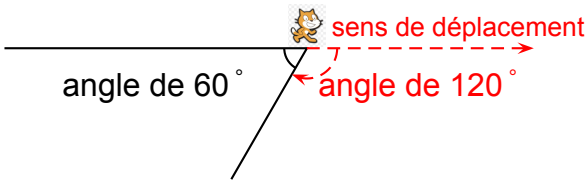
s'orienter à 90 : pour se déplacer vers la gauche

s'orienter à 90 : pour se déplacer vers le bas

Les angles

Dans le tracé ci-dessous, pour obtenir un angle de 60 °, on peut utiliser l'instruction :

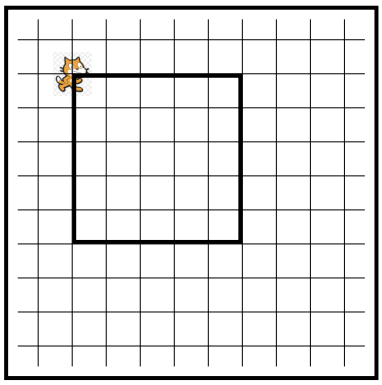
tourner de ↺ de 120 degrés



Le chat indique la position de départ.

Voici ci-contre un programme réalisé avec Scratch pour construire un parallélogramme.
Selon la longueur et l'angle donnés, ce parallélogramme peut être particulier (rectangle, losange, carré).

1. Dessiner en annexe 2 le parallélogramme obtenu avec la **longueur** et l'**angle** donnés.
2. Quelle valeur faut-il donner à **longueur** et quelle valeur à **angle** pour obtenir la figure ci-dessous ?



Le côté d'un carreau représente 20 pas.

```
quand [drapeau] est cliqué
effacer tout
s'orienter à 90
demander Donne une longueur et attendre
mettre longueur à réponse
demander Donne un angle et attendre
mettre angle à réponse
stylo en position d'écriture
avancer de 100 pas
tourner ↻ de angle degrés
avancer de longueur pas
tourner ↻ de (180 - angle) degrés
avancer de 100 pas
tourner ↻ de angle degrés
avancer de longueur pas
tourner ↻ de (180 - angle) degrés
relever le stylo
```

Script

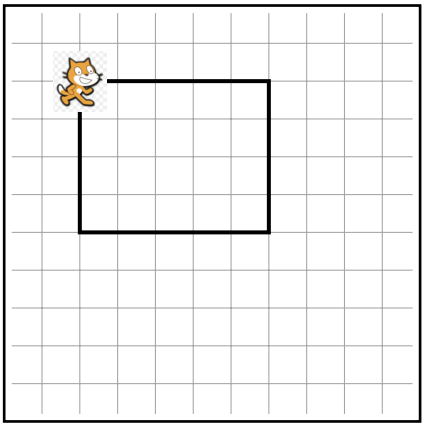
3. Un élève a choisi la longueur 50 et l'angle 75 ° puis a recopié la figure obtenue après exécution du script.
   Lequel des trois parallélogrammes ci-dessous a-t-il tracé ?
   Écrire sur la copie la lettre correspondante.

A **B** **C**

Correction

Fichier Scratch : `sg_nouvelle_caledonie_2019_mars.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/373427558`



1. En suivant le programme, une fois que le stylo est en position d'écriture, le chat est orienté vers la droite, et il avance d'abord de 100 pas, d'où un premier segment horizontal de 100 pas (donc 5 carreaux sur la figure).
   Ensuite, il tourne vers la droite, de **angle**°, soit ici 90°, donc il est maintenant orienté vers le bas.
   C'est là qu'il avance de **longueur** pas, donc, ici de 80 pas (soit un segment vertical de 4 carreaux sur la figure).
   Puis il tourne de $180 - \textbf{angle} = 180 - 90 = 90°$. Il est donc maintenant orienté vers la gauche.
   Puis, le programme continue pour terminer le parallèlogramme, qui, ici, sera un rectangle. On obtient la figure ci-contre.
2. Puisque le parallélogramme est aussi un rectangle dans ce deuxième exemple, on va également choisir **angle** = 90.

   Ce parallélogramme étant même un carré, il faut que les quatre côtés soient de même longueur, de 100 pas sur la figure, donc on choisit **longueur** = 100.
3. Sur les trois figures proposées la longueur des côtés qui ne sont pas horizontaux est bien la moitié de la longueur des côtés horizontaux, donc seul l'angle permettra de trancher.

   Avec le rappel fait au début de l'exercice, si l'angle saisit est de 75°, l'angle constaté entre les deux segments tracés sera l'angle supplémentaire, dont la mesure sera donc : $180 - 75 = 105°$.

5 Grèce, juin 2019, série générale

Références

Série : générale

Académie : Grèce

Date : juin 2019

Exercice 4 - 16 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- rosace
- déplacement
- grandeurs et mesures

Thèmes informatiques

- variable déclarée
- activité du lutin
- itération bornée
- bloc sans paramètres

Compétences travaillées

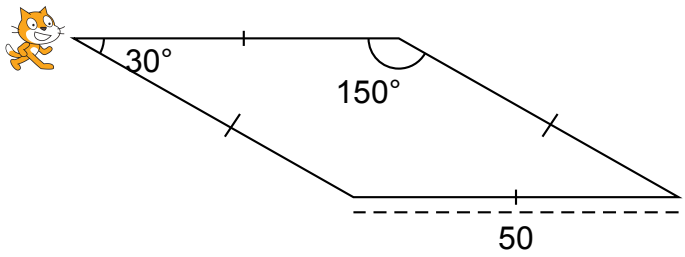
- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Grèce, juin 2019

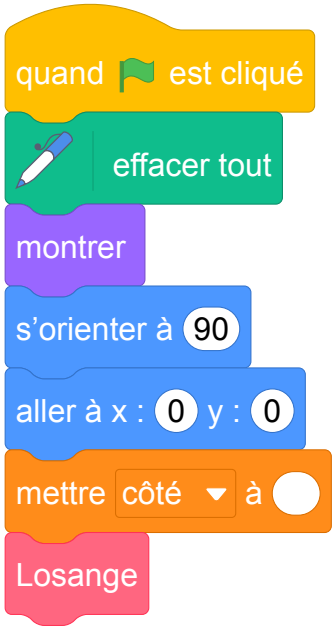
Pour occuper son petit frère, Lucie, qui aime bien l'informatique, décide de fabriquer des rosaces à colorier. Elle décide de partir d'un motif ayant la forme d'un losange.

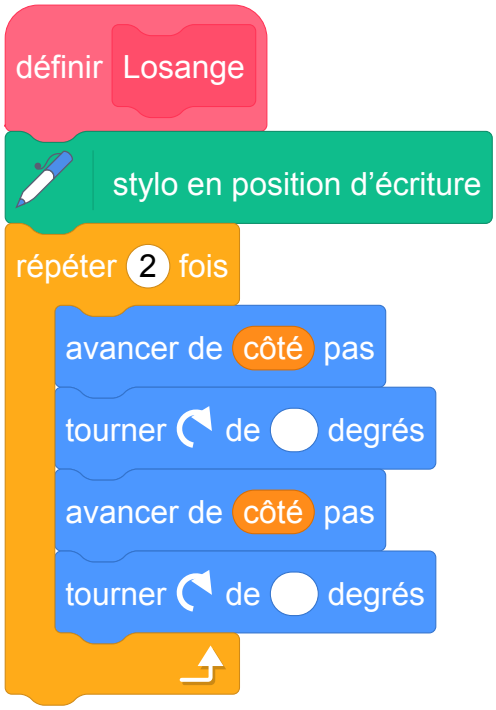
A l'aide d'un logiciel de programmation assisté (type Scratch), elle a représenté le motif suivant :



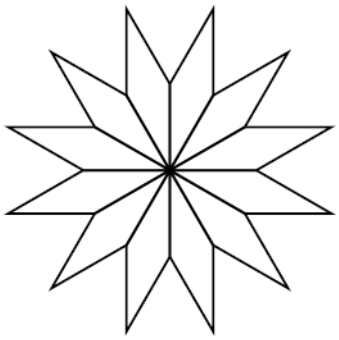
Il s'agit d'un losange dont les côtés ont pour longueur 50 pas et dont les angles aigus mesurent 30° et les angles obtus 150°.

Afin de représenter ce losange, elle a écrit le programme suivant :





1. Compléter dans l'annexe jointe le programme ci-dessus en remplaçant les pointillés par les bonnes valeurs pour que le losange soit dessiné tel qu'il est défini.
2. En utilisant le losange ci-dessus, elle obtient la rosace suivante qui n'est pas en vraie grandeur :



Quelle transformation géométrique, partant du premier losange ABCD et répétée 12 fois, a été utilisée pour obtenir cette figure ? Définir le mieux que vous pouvez cette transformation.

3. Pour finir, Lucie souhaite encore compléter cette rosace de trois façons différentes. Pour cela trois programmes ont été effectués.

   Recopier sur votre copie le numéro des trois programmes, et pour chacun, la lettre de la figure qui lui est associée.

Programme 1 :

```
quand est cliqué
effacer tout
montrer
s'orienter à 90
aller à x : 0 y : 0
mettre côté à 50
répéter 12 fois
    Losange
    tourner de 30 degrés
ajouter -25 à côté
répéter 12 fois
    Losange
    tourner de 30 degrés
cacher
```

Programme 2 :

```
quand est cliqué
effacer tout
montrer
s'orienter à 90
aller à x : 0 y : 0
mettre côté à 50
répéter 12 fois
    Losange
    tourner de 30 degrés
tourner de 15 degrés
ajouter 10 à côté
répéter 12 fois
    Losange
    tourner de 30 degrés
cacher
```

Programme 3 :

```
quand est cliqué
effacer tout
montrer
s'orienter à 90
aller à x : 0 y : 0
mettre côté à 50
répéter 12 fois
    Losange
    tourner de 30 degrés
ajouter 10 à côté
répéter 12 fois
    Losange
    tourner de 30 degrés
cacher
```

Figure A :

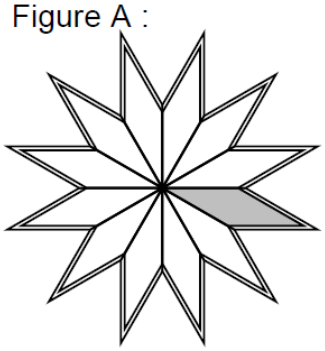


Figure B :

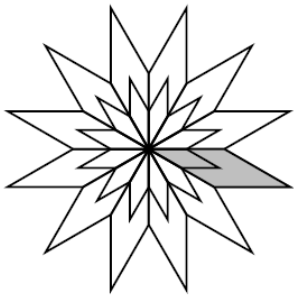
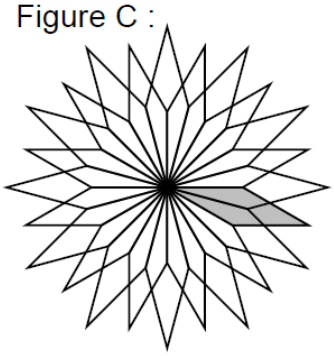


Figure C :

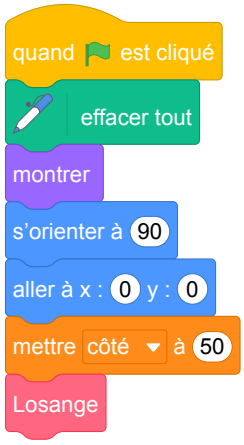


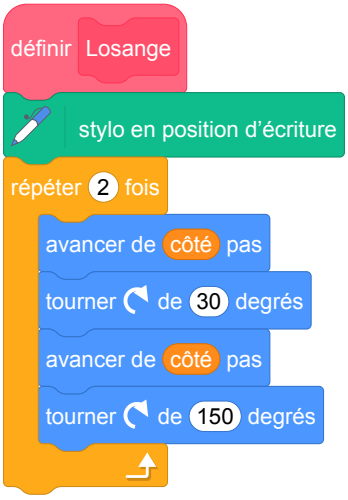
Pour plus de lisibilité, le losange initial a été grisé.

Correction

Lien vers le fichier Scratch en ligne :

- `https://scratch.mit.edu/projects/408274705`
- `https://scratch.mit.edu/projects/426493224`

1. 



2. La rotation de centre O et d'angle 30° dans n'importe quel sens répétée 12 fois permet d'obtenir la rosace à partir du losange.
3.
   Le programme 1 permet d'obtenir la figure B.
   Le programme 2 permet d'obtenir la figure C.
   Le programme 3 permet d'obtenir la figure A.

Analyse

— Cet exercice est plutôt un exercice de compréhension d'algorithme.

— Question 2, on parle du losange ABCD mais sur la figure il n'y a pas le nom du losange.

— Les instructions « montrer » et « cacher » pourraient avoir un intéret si des pauses dans le déplacement du lutin introduites avec le bloc « attendre ... »permettaient de visualiser la construction progressive de la figure. Sans pause on ne voit que la figure finale. L'utilisation de ces deux instructions est donc sans intérêt.

— La variable « côté »est une variable globale (comme toutes les variables dans Scratch) et est utilisée dans le bloc Losange sans y être définie. Cela pose des problèmes de compréhension des variables. Il aurait mieux valu utiliser un bloc avec paramètre avec en paramètre la longueur du côté du losange.

— Soit dans le bloc Losange, il faut « relever le stylo »en fin de script il faut passer l'instruction « stylo en position d'écriture » dans le programme principal. Sinon on se retrouve avec un programme qui met le stylo en position d'écriture à chaque fois que le bloc est appelé sans jamais le relever.

— Pour chacun des 3 programmes qui sont proposés on aurait pu utiliser une seule boucle « répéter 12 fois »soit en les insérant dans une boucle « répéter 2 fois », soit en créant un bloc à paramètre pour cette séquence, afin de factoriser le code. Il serait intéressant de proposer une question sur ce sujet.

Propositions

Question supplémentaire :

Modifier le programme 1 pour qu'il permette de tracer le même dessin mais en utilisant une seule boucle « répéter 12 fois ».

6 Centres étrangers, juin 2019, série générale

Références

Série : générale

Académie : Centres étrangers

Date : juin 2019

Exercice 3 - 16 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie sans transformation
- caractérisation des figures
- calcul sans programme
- périmètre

Thèmes informatiques

- variable réservée
- calcul
- itération bornée
- bloc sans parametre

Compétences travaillées

- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Centres étrangers, juin 2019

Partie I

Dans cette partie, toutes les longueurs sont exprimées en centimètres.

On considère les deux figures ci-dessous, un triangle équilatéral et un rectangle, où x représente un nombre positif quelconque.

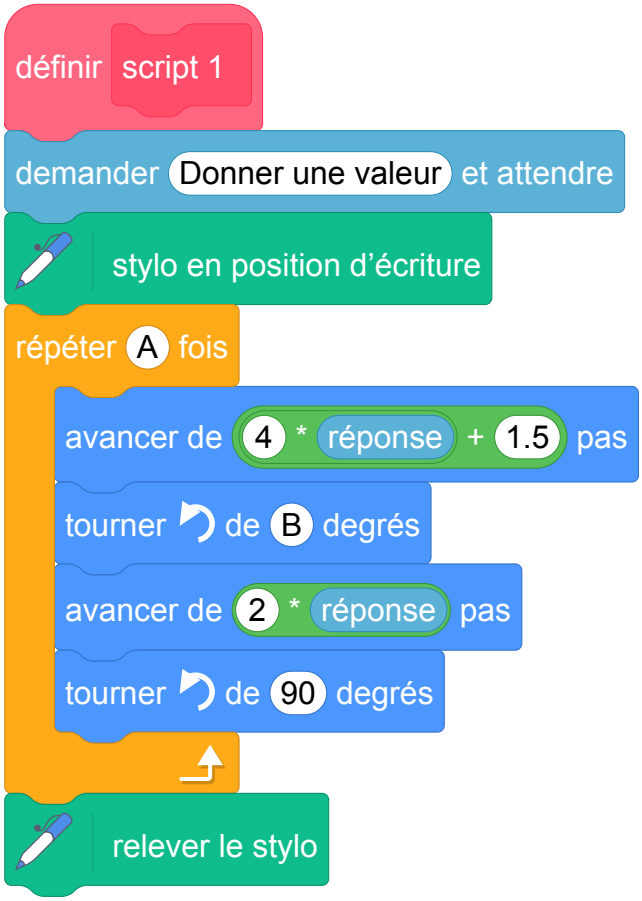
1. Construire le triangle équilatéral pour $x = 2$.
2. (a) Démontrer que le périmètre du rectangle en fonction de x peut s'écrire $12x+3$.
   (b) Pour quelle valeur de x le périmètre du rectangle est-il égal à 18 cm ?
3. Est-il vrai que les deux figures ont le même périmètre pour toutes les valeurs de x ? Justifier.

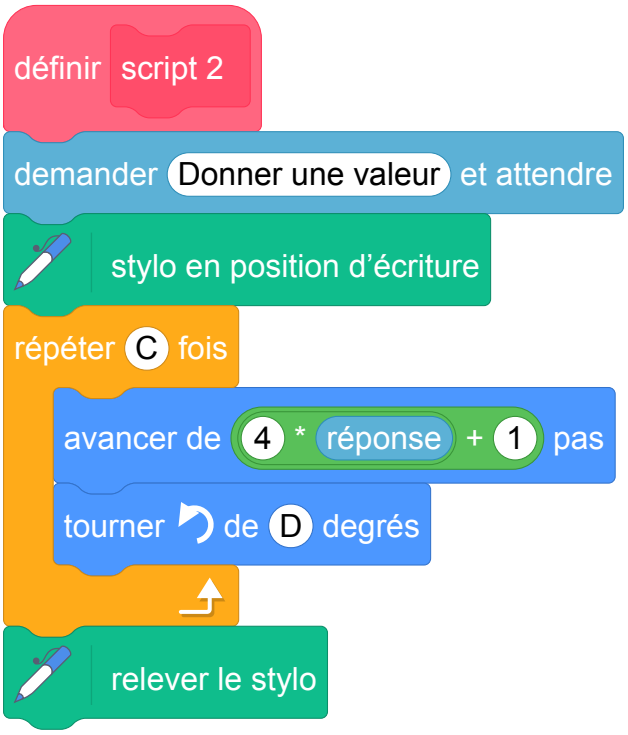
Partie II

On a créé les scripts (ci-dessous) sur Scratch qui, après avoir demandé la valeur de x à l'utilisateur, construisent les deux figures de la partie I.

Dans ces deux scripts, les lettres A, B, C et D remplacent des nombres.

Donner des valeurs à A, B, C et D pour que ces deux scripts permettent de construire les figures de la partie I et préciser alors la figure associée à chacun des scripts.





Correction

Fichiers Scratch : `sg_centres_etrangers_2019_juin.sb3`, `sg_centres_etrangers_2019_juin_correction.sb3` et `sg_centres_etrangers_2019_juin_v2.sb3`.
Liens vers le fichier Scratch sur le Studio Scratch c3i-brevet :

— https://scratch.mit.edu/projects/386321688/

— https://scratch.mit.edu/projects/386325683/

— https://scratch.mit.edu/projects/405922718/

Partie I

1. On trace un segment de longueur $4 \times 2 + 1 = 8 + 1 = 9$ cm. Par les deux extrémités de ce segment on trace deux arcs de cercle de rayon 9 (cm) qui se coupent au troisième sommet du triangle équilatéral.
2. (a) Le périmètre du rectangle est égal à :
   $2(\textit{Longueur} + \textit{largeur}) = 2(4x + 1,5 + 2x) = 2(6x + 1,5) = 12x + 3.$
   (b) Il faut résoudre l'équation donnée.

$$\begin{aligned} 12x + 3 &= 18 \\ 12x &= 15 \\ x &= \frac{15}{12} = \frac{5}{4} = 1,25. \end{aligned}$$

3. Le périmètre du triangle équilatéral est égal à : $3 \times (4x + 1) = 3 \times 4x + 3 \times 1 = 12x + 3$.
   Quel que soit le nombre positif x, le triangle équilatéral et le rectangle ont le même périmètre.

Partie II

A = 2 (on trace deux fois la longueur puis la largeur)

B = 90 (mesures des angles d'un rectangle)

C = 3 (tracé des trois côtés)

D = 120 (chaque angle d'un triangle équilatéral mesure 60 degrés).

Le premier script trace le rectangle et le second le triangle équilatéral.

Analyse

La partie I de l'exercice évalue les connaissances géométriques et le niveau de calcul littéral de l'élève ainsi que sa compréhension de la notion de démonstration. La partie II évalue la transposition de ces connaissances en langage de programmation.

L'exercice parle des scripts et non d'un programme car aucune instruction d'événements n'est proposée. Par contre, sur Scratch, il est possible d'exécuter un bloc en double cliquant dessus autant de fois souhaité.

L'élève est sensé savoir que la variable réponse contient la valeur tapée par l'utilisateur, on ne voit pas de variable du type x, côté, etc.

Propositions

Pour la partie II, afin de voir si l'élève a bien compris les scripts proposés, il aurait été mieux de travailler avec un triangle équilatéral et un carré de côtés de même longueur. Dans l'état actuel, le fait d'avoir des expressions littérales différentes pour chacune des figures géométriques permet à l'élève de savoir quel script construit quelle figure sans forcément comprendre son contenu.

Dans la partie I, il est indiqué que toutes les longueurs sont exprimées en centimètre. Pour rester cohérent, dans la partie II, il faut indiquer que toutes les longueurs sont exprimées en pas de Scratch pour la version 3.0 ou en pixel pour la version 2.0.

Les deux scripts proposés dans l'exercice demandent à l'utilisateur indépendamment une valeur qui est sensé être utilisée pour la variable *x* de la partie I. Or la variable *x* est commune aux deux figures géométriques donc elle ne doit pas avoir deux valeurs différentes. Il aurait fallu proposer un programme principal qui demande une seule fois une valeur pour la variable *x*. Les scripts peuvent soit utiliser cette variable *x* en tant que variable globale soit la recevoir en tant que paramètre (voir fichier Scratch `sg_centres_etrangers_2019_juin_v2.sb3`).

7 Antilles-Guyane, juin 2019, série générale

Références

Série : générale

Académie : Antilles-Guyanne

Date : juin 2019

Exercice 2 - 18 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- rosace
- déplacement
- grandeurs et mesures
- rotation
- proportionnalité

Thèmes informatiques

- activité du lutin
- itération bornée
- bloc sans paramètres

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Antilles-Guyane, Juin 2019

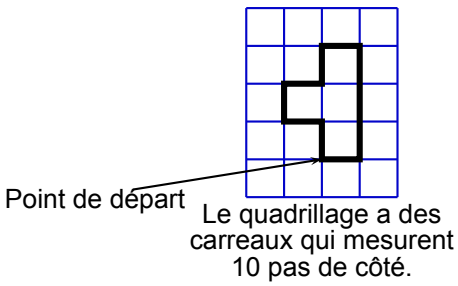
s'orienter à 90 signifie que l'on se tourne vers la droite.

Mathieu, Pierre et Elise souhaitent tracer le motif ci-dessous à l'aide de leur ordinateur. Ils commencent tous par le **script commun** ci-dessous, mais écrivent un script **Motif** différent.

Script commun aux trois élèves

```
1 quand [drapeau] est cliqué
2 aller à x : -160 y : -100
3 s'orienter à 90
4 effacer tout
5 mettre la taille du stylo à 4
6 stylo en position d'écriture
7 Motif
```

Motif



Motif de Mathieu

```
définir Motif
avancer de 10 pas
tourner ↺ de 90 degrés
avancer de 30 pas
tourner ↺ de 90 degrés
avancer de 20 pas
répéter 2 fois
    tourner ↺ de 90 degrés
    avancer de 10 pas
tourner ↻ de 90 degrés
avancer de 20 pas
```

Motif de Pierre

```
définir Motif
avancer de 10 pas
tourner ↺ de 90 degrés
avancer de 30 pas
répéter 2 fois
    tourner ↺ de 90 degrés
    avancer de 10 pas
    tourner ↺ de 90 degrés
    avancer de 10 pas
    tourner ↺ de 90 degrés
    avancer de 10 pas
tourner ↺ de 90 degrés
```

Motif d'Élise

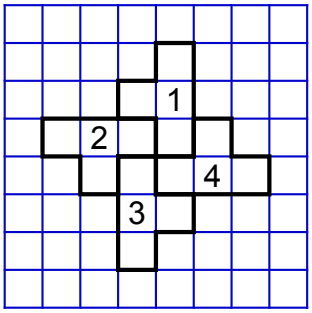
```
définir Motif
avancer de 10 pas
tourner ↺ de 90 degrés
avancer de 30 pas
répéter 2 fois
    tourner ↺ de 90 degrés
    avancer de 10 pas
    tourner ↺ de 90 degrés
    avancer de 10 pas
    tourner ↻ de 90 degrés
    avancer de 10 pas
tourner ↺ de 90 degrés
```

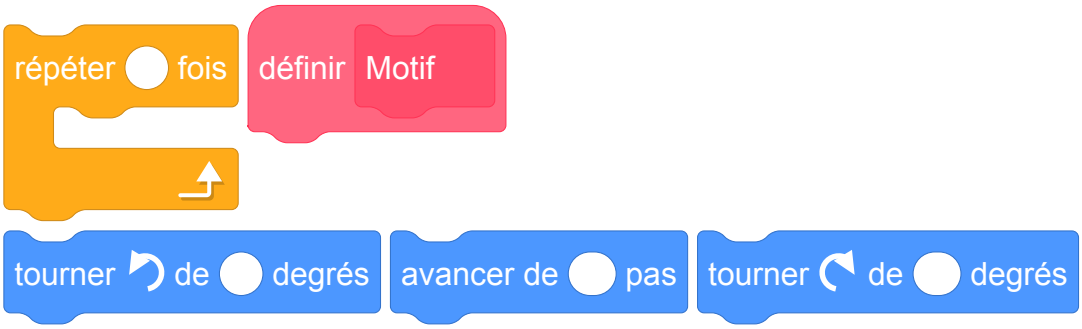
1. Tracer le motif de Mathieu en prenant comme échelle : 1 cm pour 10 pas.
2. Quel élève a un script permettant d'obtenir le motif souhaité ? On ne demande pas de justifier.

3.

a. On utilise ce motif pour obtenir la figure ci-contre. Quelle transformation du plan permet de passer à la fois du motif 1 au motif 2, du motif 2 au motif 3 et du motif 3 au motif 4 ?

b. Modifier le **script commun** à partir de la ligne 7 incluse pour obtenir la figure voulue. On écrira sur la copie uniquement la partie modifiée. Vous pourrez utiliser certaines ou toutes les instructions suivantes :

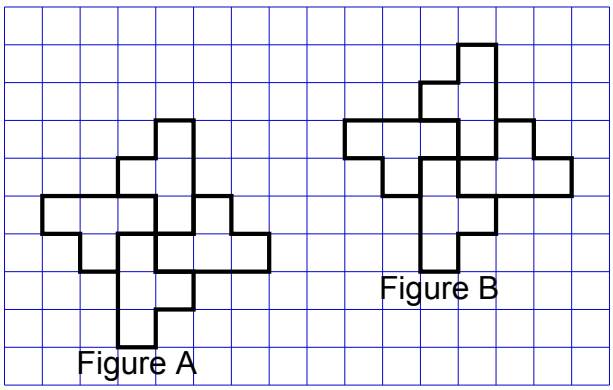




4. Un élève trace les deux figures A et B que vous trouverez en **ANNEXE**.

Placer sur cette annexe, **qui est à rendre avec la copie**, le centre O de la symétrie centrale qui transforme la figure A en figure B.

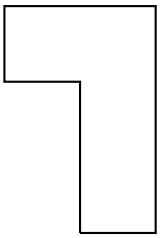
ANNEXE



Correction

Fichier Scratch : `sg_sg_antilles_guyane_2019_juin.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/773071949`

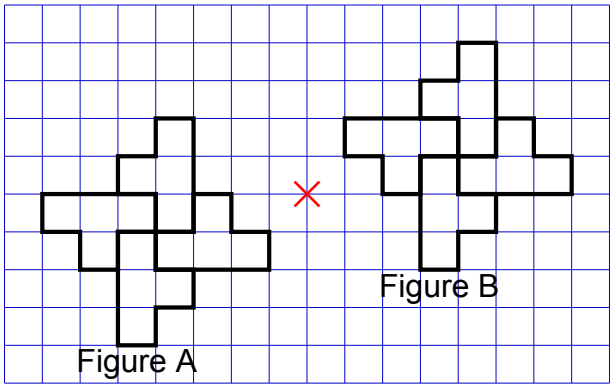
1. Le motif de Mathieu en prenant comme échelle : 1 cm pour 10 pas est :



2. C'est le script d'Élise qui permet d'obtenir le motif souhaité.
3. (a) La rotation centrée au point commun des quatre motifs (au centre de la figure et de $+90^\circ$) permet de passer de 1 à 2 de 2 à 3 de 3 à 4.

   (b)
   ```
   7 répéter 4 fois
   8     Motif
   9     tourner ↺ de 90 degrés
   ```
4. Voir sur l'annexe le point en rouge.

ANNEXE



8 Métropole, juillet 2019, série générale

Références

Série : générale

Académie : Métropole-La Réunion

Date : juillet 2019

Exercice 4 - 19 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- frise
- déplacement
- translation
- probabilités

Thèmes informatiques

- activité du lutin
- aléatoire
- branchement conditionnel
- itération bornée
- bloc sans parametre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	1
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	1
itération conditionnelle	0
imbrication de structures	1
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	1
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole, Juillet 2019

On veut réaliser un dessin constitué de deux types d'éléments (tirets et carrés) mis bout à bout.
Chaque script ci-contre trace un élément, et déplace le stylo.

On rappelle que s'orienter à 90 signifie qu'on oriente le stylo vers la droite.

```
définir carré
s'orienter à 90
tourner de ↺ de 90 degrés
répéter 4 fois
    avancer de 5 pas
    tourner ↻ de 90 degrés
    avancer de 5 pas
relever le stylo
s'orienter à 90
avancer de 10 pas
stylo en position d'écriture
```

```
définir tiret
s'orienter à 90
avancer de 10 pas
```

1. En prenant 1 cm pour 2 pas, représenter la figure obtenue si on exécute le script carré.

   Préciser les positions de départ et d'arrivée du stylo sur votre figure.

   Pour tracer le dessin complet, on a réalisé 2 scripts qui se servent des blocs carré et tiret ci-dessus :

Script 1

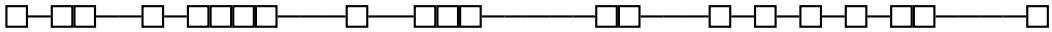
```
quand flèche haut est pressée
aller à x : -230 y : 0
s'orienter à 90
effacer tout
stylo en position d'écriture
répéter 23 fois
    carré
    tiret
```

Script 2

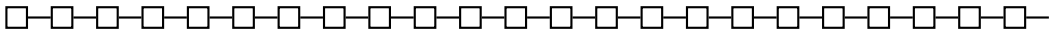
```
1  quand flèche bas est pressée
2  aller à x : -230 y : 0
3  s'orienter à 90
4  effacer tout
5  stylo en position d'écriture
6  répéter 46 fois
7      si nombre aléatoire entre 1 et 2 = 1 alors
8          carré
9      sinon
10         tiret
```

On exécute les deux scripts et on obtient les deux dessins ci-dessous.

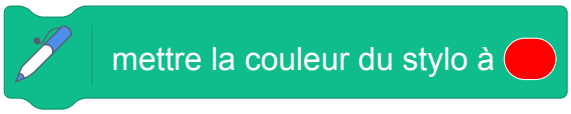
Dessin A

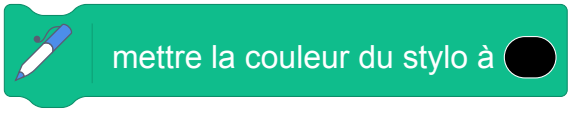


Dessin B



2. Attribuer à chaque script la figure dessinée. Justifier votre choix.
3. On exécute le script 2.
   (a) Quelle est la probabilité que le premier élément tracé soit un carré ?
   (b) Quelle est la probabilité que les deux premiers éléments soient des carrés ?
4. Dans le script 2, on aimerait que la couleur des différents éléments, tirets ou carrés, soit aléatoire, avec à chaque fois 50 % de chance d'avoir un élément noir et 50 % de chance d'avoir un élément rouge.
   Écrire la suite d'instructions qu'il faut alors créer et préciser où l'insérer dans le script 2.

   Indication : on pourra utiliser les instructions 

   et 

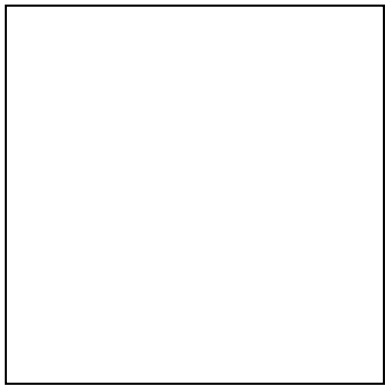
   pour choisir la couleur du stylo.

Correction

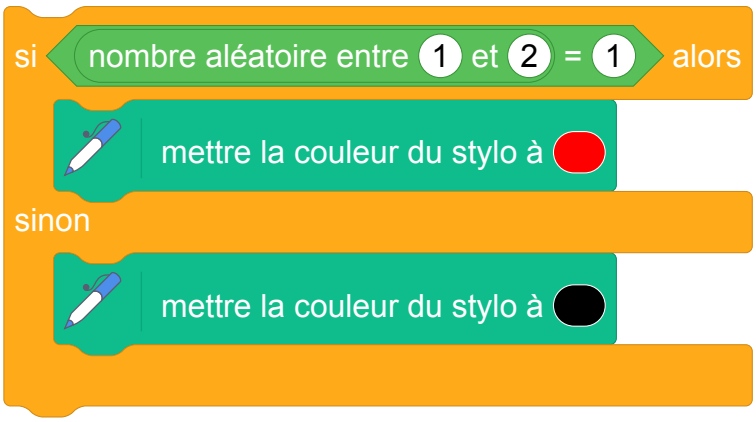
Fichier Scratch : `sg_metropole_2019_juillet.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/773601774/`

1. Le script carré trace un carré en traçant 4 fois deux demi-côtés de 5 pas formant un angle de 90°, donc chaque côté du carré correspond à 10 pas, donc à 5 cm.



2. Le script 1 dessine 23 fois un carré suivi d'un tiret donc ne peut être que le dessin B.
   Le script 2 dessine 46 fois de manière aléatoire un carré ou un tiret et pourrait être le dessin A ou le dessin B mais comme le script 1 ne peut donner que le dessin B le script 2 a forcément donné le dessin A.
3. (a) En exécutant le script 2, le premier élément tracé est un carré si le nombre aléatoire prend la valeur 1 qui est l'une des deux valeurs possibles. La probabilité est donc de $\frac{1}{2} = 0,5$.
   (b) Pour les deux premiers éléments dessinés, il y a 4 possibilités équiprobables :
   carré – carré ; carré – tiret ; tiret – carré ; tiret – tiret.
   La probabilité que les deux premiers éléments dessinés soient des carrés est $\frac{1}{4}$, soit $0,25$.
   On peut aussi considérer le fait qu'il y a une chance sur deux de tracer un carré à la première itération et aussi une chance sur deux à la deuxième. On a donc une probabilité de $\frac{1}{2} \times \frac{1}{2} = \frac{1}{4} = 0,25$ que les deux premiers éléments dessinés soient des carrés.
4. Entre les lignes 6 et 7 du script 2, on peut insérer :



Analyse

Dans ce sujet, on travaille la compréhension et la modification d'algorithme (et les probabilités en question 2).

Si la nature de l'objet à dessiner à la première question est évidente en raison du nom du bloc, le fait que le bloc « carré »ne soit pas défini comme d'habitude peut dérouter les élèves. Ils vont devoir comprendre comment la figure est tracée pour en déterminer les dimensions, information nécessaire pour traiter la première question.

La réponse à la question 2 est également évidente puisqu'un seul des deux scripts a une instruction indiquant un comportement aléatoire du lutin.

Par contre dans la question 4 on demande aux élèves d'avoir compris cette partie du script (le bloc « Si ... Alors ... Sinon ») pour le transposer afin de fixer aléatoirement la couleurs des objets. Cela demande aussi une bonne compréhension de la boucle bornée, pour déterminer où placer ce bloc.

9 Amérique du sud, novembre 2019, série générale

Références

Série : générale

Académie : Amérique du sud

Date : novembre 2019

Exercice 6 - 16 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- frise
- déplacement
- grandeurs et mesures
- translation

Thèmes informatiques

- activité du lutin
- itération bornée
- bloc sans paramètres

Compétences travaillées

- comprendre un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Amérique du sud, novembre 2019

Voici les copies d'écran d'un programme qui permet d'obtenir une frise.

Script de la frise	Bloc motif	Pour information
quand est cliqué s'orienter à 90 aller à x : -150 y : 0 effacer tout répéter 4 fois Motif avancer de 30 pas	définir Motif stylo en position d'écriture avancer de 20 pas tourner de 60 degrés avancer de 40 pas tourner de 120 degrés avancer de 40 pas s'orienter à 90 relever le stylo	L'instruction s'orienter à 90 signifie qu'on s'oriente vers la droite.

Frise obtenue avec le script

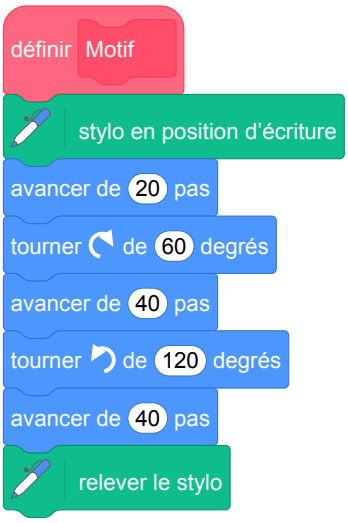
Point de départ

1. Quelle distance le lutin a-t-il parcourue pour tracer **un seul motif** de la frise ?
2. On modifie le programme, dans cette question seulement :
   - on ne modifie pas le script de la frise.
   - dans le bloc motif, il enlève l'instruction : relever le stylo

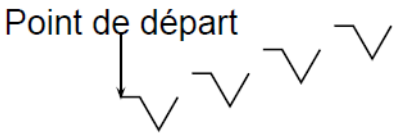
   Dessiner à main levée la frise obtenue avec ce nouveau programme.

3. On utilise maintenant le bloc motif ci-dessous. Laquelle des deux frises obtient-il ? Expliquer pourquoi.

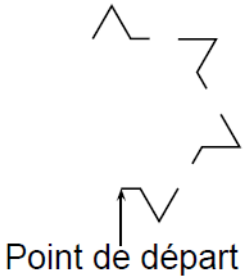
Bloc motif modifié



Frise n° 1 est :



Frise n° 2 est :



Correction

Fichier Scratch : `sg_amerique-du-sud_2019_novembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/481409943/`

1. Pour un motif le lutin parcourt :
   $20 + 40 + 40 = 100$ pixels.
2. On obtient le dessin continu suivant :

3. On obtient la frise n° 2 : avec le nouveau motif à la fin de son exécution on ne remets pas l'orientation à 90 contrairement à la frise n° 1.

Analyse

Les questions 1 et 2 sont triviales. La question 1 ne sert qu'à découvrir le motif pour être capable de traiter la question 3.

La question 3 nécessite d'être capable d'exécuter le programme pas à pas pour déterminer le changement d'orientation d'un motif au suivant.

Propositions

Il serait plus intéressant de demander à l'élève :

— de dessiner le motif

— de déterminer la largeur du motif (sur l'axe horizontal). Cela lui nécessiterait de comprendre que sur une partie le motif suit le tracé d'un triangle équilatéral et d'en connaître les propriétés (compétence de mathématiques).

10 Nouvelle-Calédonie, décembre 2019, série générale

Références

Série : générale

Académie :Nouvelle-Calédonie

Date : décembre 2019

Exercice 8 - 13 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- spirale
- grandeurs et mesures
- rotation

Thèmes informatiques

- variable déclarée
- activité du lutin
- itération bornée

Compétences travaillées

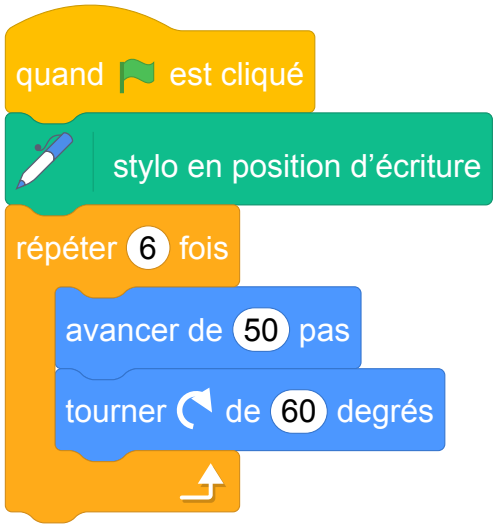
- comprendre un algorithme
- compléter un algorithme
- écrire un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

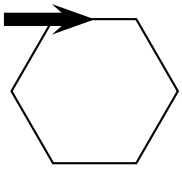
Énoncé - Nouvelle-Calédonie, Décembre 2019

Dans les figures de cet exercice la flèche indique la position et l'orientation du lutin au départ.

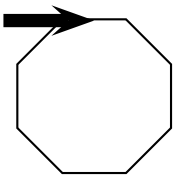
1. Indiquer sur la copie le numéro du dessin correspondant au script ci-dessous.



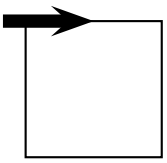
Dessin n° 1



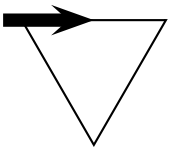
Dessin n° 2



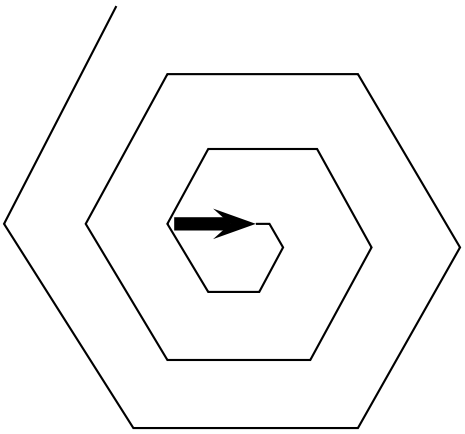
Dessin n° 3



2. Sur l'**annexe**, compléter les deux informations manquantes du script qui permet de réaliser la figure ci-dessous

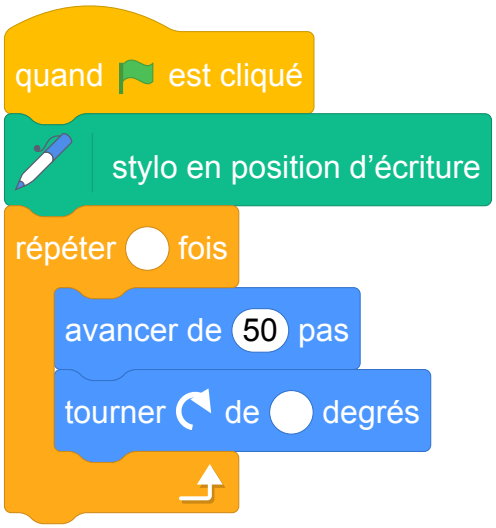


3. En ordonnant les instructions proposées en **annexe 2**, compléter le script permettant de réaliser la figure ci-dessous. On indiquera les numéros des instructions sur l'annexe.



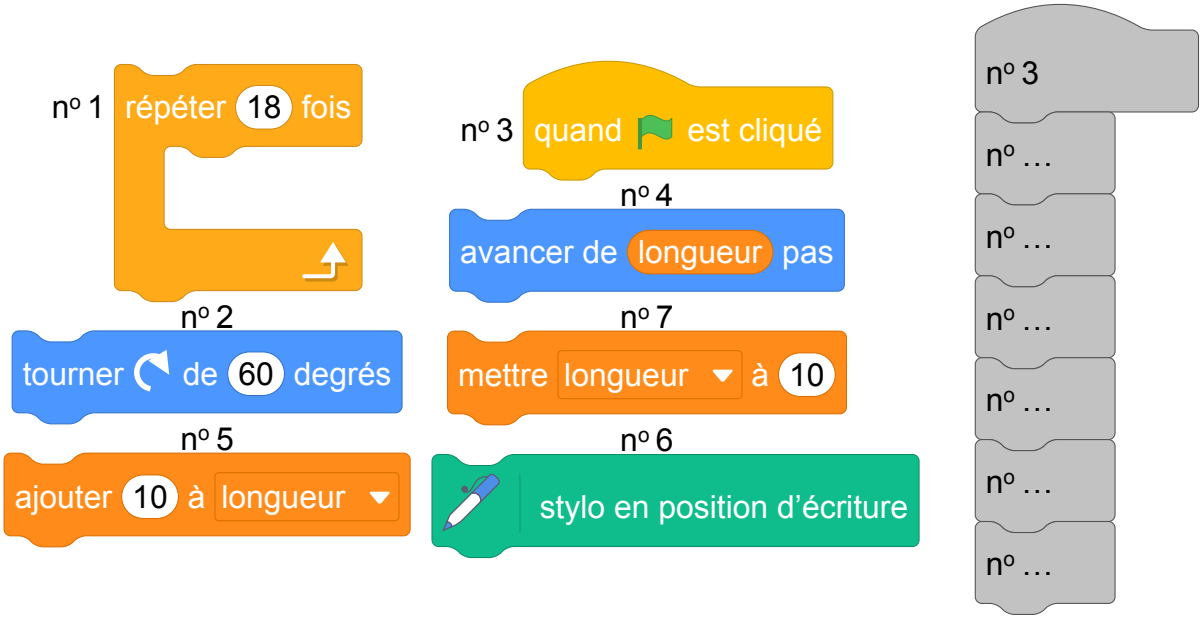
Annexe

Question 2



Question 3

Pour ce script on a créé la variable longueur. Compléter en mettant les numéros à leur place.



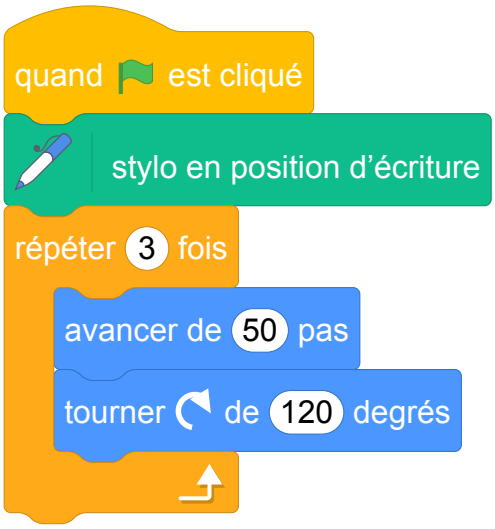
Correction

Fichier Scratch : `sg_nouvelle_caledonie_2019_decembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/481421869/`

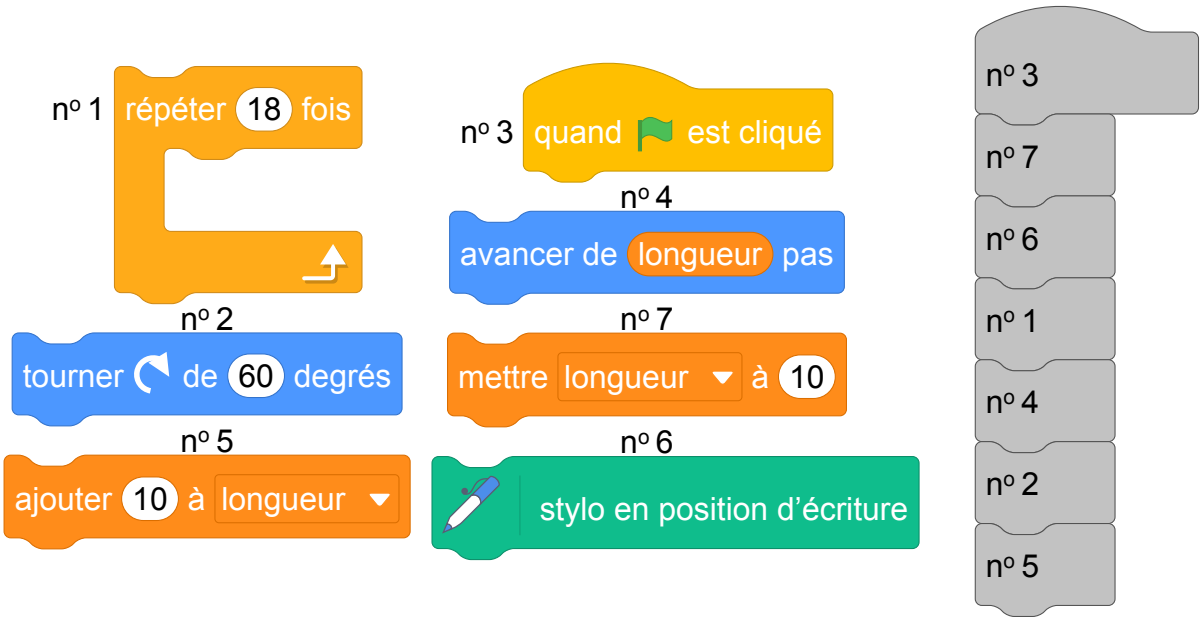
1. La figure obtenue a six côtés : c'est le dessin n° 1 qui est obtenu.
2. Voir l'annexe.
3. Voir l'annexe.

Annexe

Question 2



Question 3



11 Amérique du Nord, juin 2019, série générale

Références

Série : générale

Académie : Amérique du Nord

Date : juin 2019

Exercice 4 - 10 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- repérage

Thèmes informatiques

- variable réservée
- activité du lutin
- branchement conditionnel
- évènement sans message

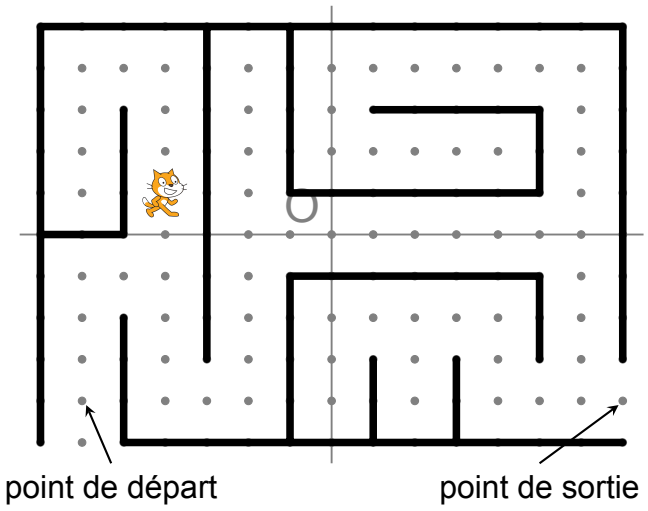
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	1
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	1
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	1
événements internes	1
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Amérique du Nord, juin 2019

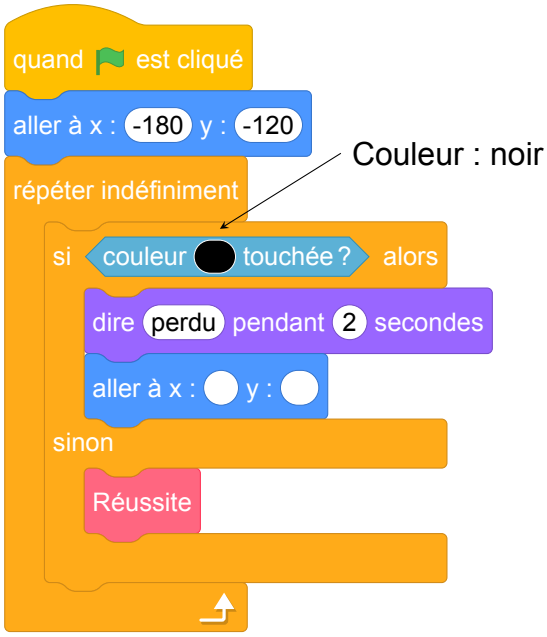
On a programmé un jeu. Le but du jeu est de sortir du labyrinthe. Au début du jeu, le lutin se place au point de départ. Lorsque le lutin touche un mur, représenté par un trait noir épais, il revient au point de départ.



L'arrière-plan est constitué d'un repère d'origine O avec des points espacés de 30 pas verticalement et horizontalement.

Dans cet exercice, on considèrera que seuls les murs du labyrinthe sont noirs.

Voici le programme :



Le bloc Réussite correspond à un sous-programme qui fait dire « Gagné ! » au lutin lorsqu'il est situé au point de sortie ; le jeu s'arrête alors.

1. Recopier et compléter l'instruction aller à x : y : du programme pour ramener le lutin au point de départ si la couleur noire est touchée.
2. Quelle est la distance minimale parcourue par le lutin entre le point de départ et le point de sortie ?
3. On lance le programme en cliquant sur le drapeau. Le lutin est au point de départ. On appuie brièvement sur la touche ↑ (« flèche haut ») puis sur la touche → (« flèche droite »). Quelles sont toutes les actions effectuées par le lutin ?

Correction

Fichier scratch : `sg_amerique_du_nord_2019_juin.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet :
`https://scratch.mit.edu/projects/408267145`

1. aller à x : -180 y : -120
2. Le chemin le plus court : monter de 3, aller à droite de 2, descendre de 3, aller à droite de 2, monter de 4, aller à droite de 8, descendre de 4, aller à droite de 1, donc en tout 27 déplacements de 30 pas soit 810 pas.
3. Le lutin monte de 30 pas puis se déplace vers la droite de 30 pas. Il percute le mur. le jeu annonce « Perdu » et replace le lutin au point de départ.

12 Asie, juin 2019, série générale

Références

Série : générale

Académie : Asie

Date : juin 2019

Exercice 3 - 17 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- repérage
- déplacement

Thèmes informatiques

- scène

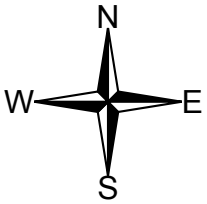
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

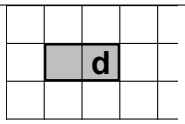
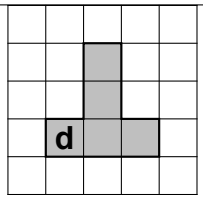
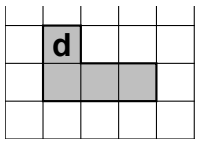
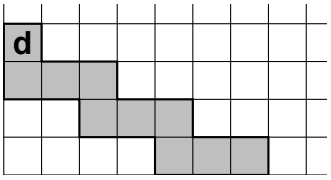
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	1
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	1
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Asie, juin 2019

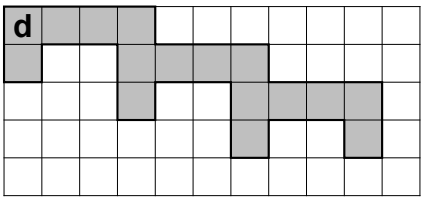
Un programme permet à un robot de se déplacer sur les cases d'un quadrillage. Chaque case atteinte est colorée en gris. Au début d'un programme, toutes les cases sont blanches, le robot se positionne sur une case de départ indiquée par un « **d** » et la colore aussitôt en gris.



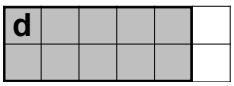
Voici des exemples de programmes et leurs effets :

• 1W	Le robot avance de 1 case vers l'ouest.	
• 2E 1W 2N	Le robot avance de 2 cases vers l'est, puis de 1 case vers l'ouest, puis de 2 cases vers le nord.	
• 3 (1S 2E)	Le robot répète 3 fois le déplacement suivant : « avancer de 1 case vers le sud puis de 2 cases vers l'est ». Soit 3 fois : 	

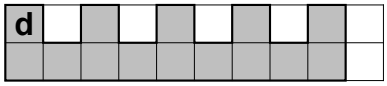
1. Voici un programme :
   Programme : 1W 2N 2E 4S 2W
   On souhaite dessiner le motif obtenu avec ce programme.
   Sur votre copie, réaliser ce motif en utilisant des carreaux, comme dans les exemples précédents.
   On marquera un « **d** » sur la case de départ.
2. Voici deux programmes :
   Programme n° 1 : 1S 3(1N 3E 2S)
   Programme n° 2 : 3(1S 1N 3E 1S)
   (a) Lequel des deux programmes permet d'obtenir le motif ci-contre ?
   (b) Expliquer pourquoi l'autre programme ne permet pas d'obtenir le motif ci-contre.



3. Voici un autre programme :
   Programme n° 3 : 4(1S 1E 1N)
   Il permet d'obtenir le résultat suivant :

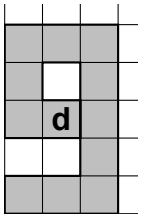


   Réécrire ce programme n° 3 en ne modifiant qu'une seule instruction afin d'obtenir ceci :



Correction

1. Motif obtenu :



2. (a) C'est le programme 2 (le programme 1 grise une case de trop).
   (b) Dans le programme 2 les deux premières instructions s'annulent, donc on ne fait qu'aller 3 fois à droite et une fois vers le bas.
3. **Programme nº 3** : 4(1S 2E 1N)

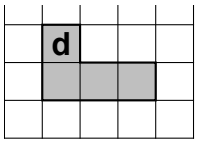
Analyse

C'est un des rares exercices d'informatique du DNB où le Scratch n'est pas utilisé. L'exercice propose de travailler sur les déplacements d'un robot sur un quadrillage en suivant des commandes simples. Puisque le système n'est pas connu par les élèves, contrairement à Scratch, il y a une partie importante d'explications. L'exercice évalue les compétences de compréhension des élèves devant un texte explicatif.

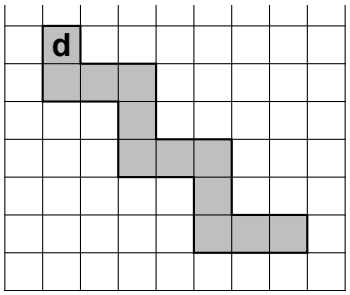
Les directions de la boussole sont indiqués avec des majuscules correspondants aux mots en anglais (N : north, E : est, S : south, W :west). Même si les majuscules N, E et S correspondent aux premières lettres des directions en français, la lettre W peut inutilement confondre les élèves. Il aurait été préférable d'utiliser la lettre O pour "ouest".

D'après les explications du début de l'exercice, dans le quadrillage, la lettre **d** correspond à la case de départ du robot. Cet emplacement peut être confondu avec la position actuelle du robot qui lui en fait n'est pas indiqué sur les images.

Dans la troisième ligne des explications du système, le schéma fournit porte à confusion. Le motif



répété trois fois donnerait le motif suivant :



et non le motif proposé dans l'exercice. La tournure des phrases est à revoir pour plus de clarté.

Dans la question 1, les élèves peuvent se demander où mettre la case de départ **d**. Il faut explicitement indiquer que l'emplacement de la case de départ n'influence pas le motif décrit par le programme proposé. Il peut être envisagé de proposer un annexe à rendre avec la copie, sur laquelle il y a déjà une case de départ.

Dans la question 3, on demande de réécrire le programme 3 en ne modifiant qu'une seule instruction. Or la définition d'une instruction dans ce système n'est pas définie : est-ce qu'on veut dire une lettre ? un nombre ? un déplacement ? La réponse attendue est que le programme 4(1S 1E 1N) doit être modifié pour obtenir 4(1S 2E 1N) c'est-à-dire on ajoute une déplacement (ou 4 déplacements selon les points de vue) vers l'est. Cette restriction d'une seule instruction est inutile. On peut très bien donner le programme 3 pour information et demander à le modifier pour obtenir le deuxième motif.

Propositions

L'exercice accessible depuis le lien suivant est proposé par l'IREM de Marseille : `https://ens-info.irem.univ-mrs.fr/wp-content/uploads/Asie-juin-2019-modifi%C3%A9-1.pdf`

13 Nouvelle Calédonie, mars 2019, série professionnelle

Références

Série : professionnelle

Académie : Nouvelle Calédonie

Date : mars 2019

Exercice 6 - 12 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- repérage
- déplacement

Thèmes informatiques

- activité du lutin

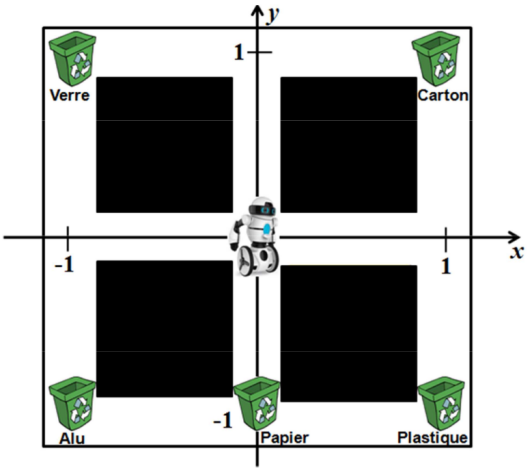
Compétences travaillées

- comprendre un algorithme

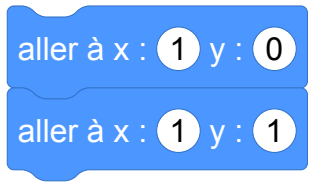
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Nouvelle Calédonie, Mars 2019 série professionnelle

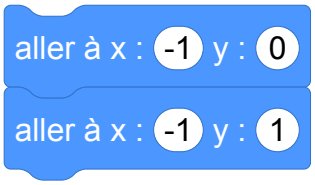
Un petit robot « Neto » fait le tri de déchets et doit jeter les canettes en aluminium dans la poubelle « Alu » d'un centre de tri. On représente ci-dessous le centre de tri dans un repère d'unité 1. Neto se trouve au point de coordonnées $(0;0)$ du centre de tri et ne peut se déplacer que sur les chemins blancs pour accéder aux différentes poubelles.



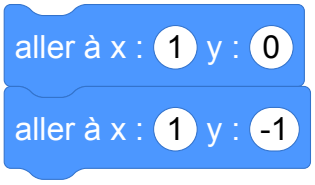
1. Relier par un trait chaque bloc d'instructions permettant à Neto d'accéder à la poubelle correspondante. On suppose que Neto part toujours de l'origine du repère $(0;0)$.



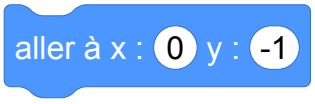
☐ ☐ Poubelle "papier"



☐ ☐ Poubelle "plastique"

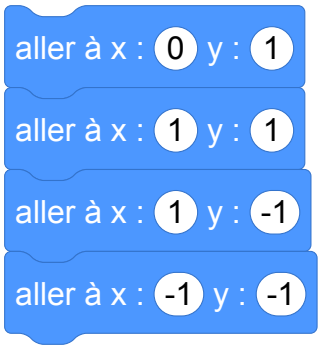


☐ ☐ Poubelle "verre"



☐ ☐ Poubelle "carton"

Pour aller à la poubelle « Alu » Neto a fait un détour en suivant les instructions suivantes :



2. Indiquer devant quelle(s) poubelle(s) Neto est passé avant d'arriver à la poubelle « Alu ».

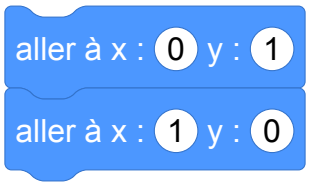
Correction

1. Les blocs d'instruction dans l'ordre sont : poubelle « carton », poubelle « verre », poubelle « plastique », poubelle « papier ».
2. Neto est passé devant : poubelle « carton », poubelle « plastique », poubelle « papier » avant d'arriver à « alu ».

Analyse

L'exercice met presque uniquement en jeu des connaissances mathématiques, notamment le repérage dans le carré unité du plan orthonormé.

La seule notion spécifique à la programmation évaluée dans cet exercice est la compréhension de l'effet d'une suite d'instructions, mais dans la question 1, lire la dernière instruction de chaque séquence suffirait à répondre. Il aurait été possible d'utiliser des suites d'instructions de déplacement relatif (« avancer », « tourner »), ou d'introduire des suites d'instructions erronées guidant le robot hors des zones autorisées, par exemple :



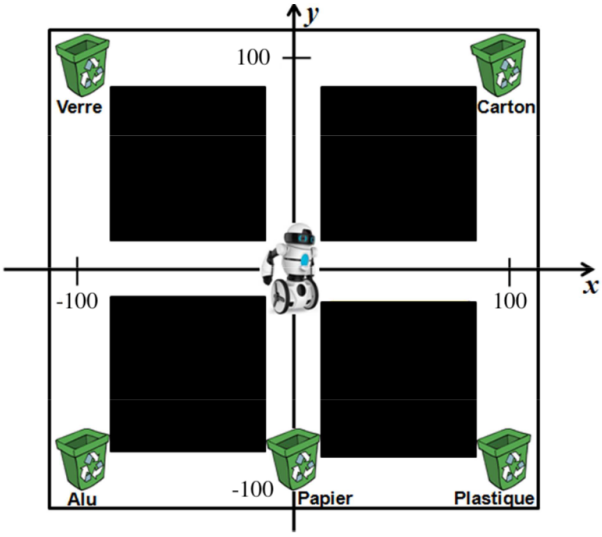
La question 2 est plus riche et demande de simuler pas à pas l'effet de chaque déplacement.

Propositions

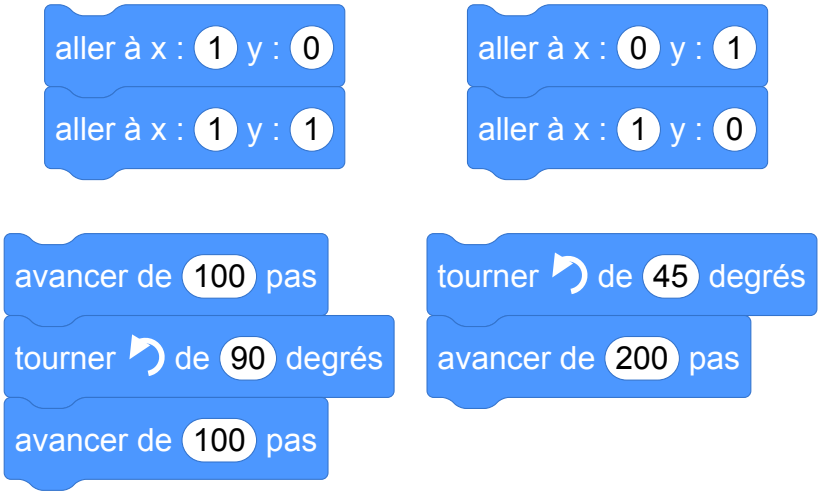
Dans la proposition suivante, on donne un exemple de reformulation de l'énoncé et de la question 1 afin de mettre davantage en jeu la compréhension de l'effet d'une séquence d'instructions, ainsi que le mode de repérage relatif des déplacements.

Proposition de modification de l'énoncé

Un petit robot « Neto » fait le tri de déchets et doit jeter les canettes en aluminium dans la poubelle « Alu » d'un centre de tri. On représente ci-dessous le centre de tri comme un carré de côté 200 centré en $(0;0)$ dans un repère cartésien. Neto se trouve au point de coordonnées $(0;0)$ du centre de tri et ne peut se déplacer que sur les chemins blancs pour accéder aux différentes poubelles.



1. On veut connaître l'effet des blocs d'instructions suivants afin de programmer le robot :



(a) Pour chacun des blocs d'instructions ci-dessus, indiquer si le robot se déplace correctement, c'est-à-dire sans sortir des chemins blancs.

(b) Pour chacun des blocs d'instructions corrects, indiquer si le robot s'arrête devant l'une des poubelles (et laquelle) ou s'il s'arrête dans une zone vide.

Brevets 2020

1 Nouvelle-Calédonie, février 2020, série générale, exercice 2

Références

Série : générale

Académie :Nouvelle-Calédonie

Date : février 2020

Exercice 2 - 12 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul
- grandeurs et mesures

Thèmes informatiques

- variable déclarée
- calcul
- capteurs

Compétences travaillées

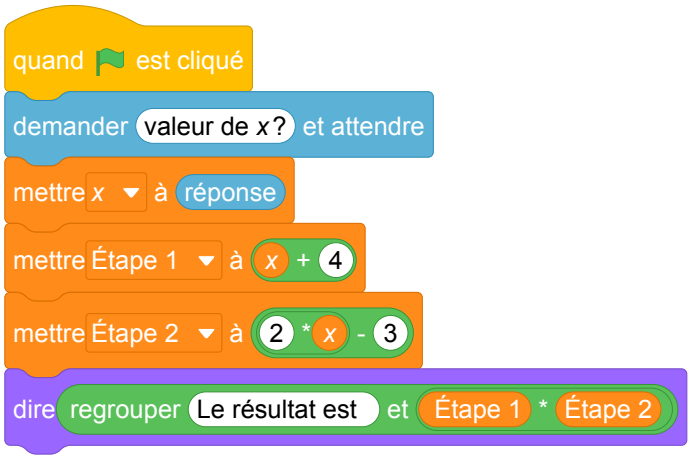
- Aucune compétence algorithmique

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Nouvelle-Calédonie, Février 2020, exercice 2

Laura a créé trois variables puis elle a réalisé le script ci-dessous.

Créer une variable



1. Vérifier que si la valeur de x est 5 alors le résultat est 63.
2. Quel résultat obtient-on si la valeur de x est -3 ?
3. Parmi les expressions suivantes, recopier celle qui correspond au programme de calcul donné par le script.

$$A = (x+4) \times (2x-3) \qquad B = x + 4 \times 2x - 3 \qquad C = x + 4 \times (2x-3)$$

4. Pour quelle(s) valeur(s) de x obtient-on un résultat égal à 0 ?

Correction

Fichier Scratch : `sg_nouvelle_caledonie_2020_fevrier_ex2.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/481463832/`

1. Si $x = 5$, Étape 1$= 5 + 4 = 9$ et Étape 2$= 2 \times 5 - 3 = 10 - 3 = 7$.
   Le résultat est donc $9 \times 7 = 63$
2. Si $x = -3$, Étape 1$= -3 + 4 = 1$ et Étape 2$= 2 \times (-3) - 3 = -6 - 3 = -9$.
   Le résultat est donc $1 \times (-9) = -9$
3. L'expression qui correspond au programme de calcul donné par le script est :

$$A = (x + 4) \times (2x - 3)$$

4. Il faut résoudre l'équation $(x + 4) \times (2x - 3) = 0$
   Or un produit est nul si et seulement si un des facteurs est nul. On a donc :
   $(x + 4) = 0$ ou $(2x - 3) = 0$
   On en déduit que $x = -4$ ou $2x = 3$
   et enfin $x = -4$ ou $x = 3 \div 2 = 1{,}5$

Analyse

Cet exercice n'a pas d'intérêt algorithmique. Encore une fois c'est un prétexte à exercice de calcul littéral !
L'écriture des variables ne correspond pas à scratch. Il aurait fallu écrire

La variable x n'est pas écrit de la même façon dans le problème...

2 Antilles-Guyane, septembre 2020, série générale

Références

Série : générale

Académie : Antilles-Guyanne

Date : septembre 2020

Exercice 5 - 23 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- probabilités
- statistiques
- simulations
- proportionnalité

Thèmes informatiques

- variable déclarée
- aléatoire
- branchement conditionnel
- itération bornée
- bloc sans paramètre

Compétences travaillées

- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	1
calcul	1
aléatoire	1
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	1
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	1
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Antilles-Guyane, septembre 2020

On dispose de deux urnes :

— une urne bleue contenant trois boules bleues numérotées : 2, 3 et 4.

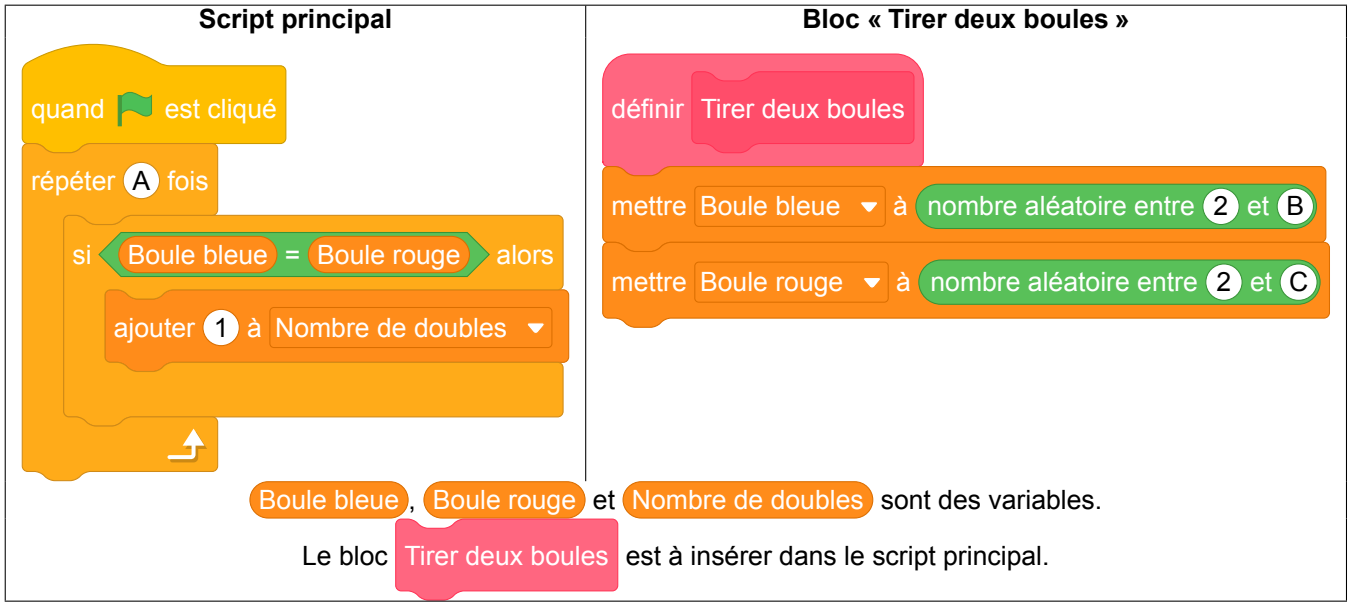
— une urne rouge contenant quatre boules rouges numérotées : 2, 3, 4 et 5.

Dans chaque urne, les boules sont indiscernables au toucher et ont la même probabilité d'être tirées.

Urne bleue	Urne rouge
2 3 4	2 3 4 5

On s'intéresse à l'expérience aléatoire suivante :
« On tire au hasard une boule bleue et on note son numéro, puis on tire au hasard une boule rouge et on note son numéro. »
Exemple : si on tire la boule bleue numérotée 3, puis la boule rouge numérotée 4, le tirage obtenu sera noté (3 ; 4).
On précise que le tirage (3 ; 4) est différent du tirage (4 ; 3).

1. On définit les deux évènements suivants :
   « On obtient deux nombres premiers » et « La somme des deux nombres est égale à 12 »
   (a) Pour chacun des deux évènements précédents, dire s'il est possible ou impossible lorsqu'on effectue l'expérience aléatoire.
   (b) Déterminer la probabilité de l'évènement « On obtient deux nombres premiers ».
2. On obtient un « double » lorsque les deux boules tirées portent le même numéro.
   Justifier que la probabilité d'obtenir un « double » lors de cette expérience, est $\frac{1}{4}$.
3. Dans cette question, aucune justification n'est attendue.
   On souhaite simuler cette expérience 1 000 fois.
   Pour cela, on a commencé à écrire un programme, à ce stade, encore incomplet. Voici des copies d'écran :



   (a) Par quels nombres faut-il remplacer les lettres A, B et C ?

(b) Dans le script principal, indiquer où placer le bloc Tirer deux boules

(c) Dans le script principal, indiquer où placer le bloc mettre Nombre de doubles à 0

(d) On souhaite obtenir la fréquence d'apparition du nombre de « doubles » obtenus.
Parmi les instructions ci-dessous, laquelle faut-il placer à la fin du script principal après la boucle « répéter » ?

Proposition 1	Proposition 2	Proposition 3
dire Nombre de doubles	dire Nombre de doubles / 1000	dire Nombre de doubles / 2

Correction

Fichier Scratch : `sg_sg_antilles_guyane_2020_septembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/481464151/`

1. (a)
   - Il est possible de tirer deux nombres premiers : (2 ; 2), (2 ;3), (2 ; 5), (3 ; 2), (3 ; 3), (3 ; 5).
   - La somme la plus grande est $4+5=9$. 12 est donc impossible à atteindre.

   (b) Il y a $3\times4=12$ tirages différents et on a vu qu'il y en avait 6 donnant deux nombres premiers. La probabilité est donc égale à $\frac{6}{12}=\frac{1}{2}=0{,}5$.
2. On peut obtenir les doubles (2 ; 2), (3 ; 3) et (4 ; 4), donc 3 doubles sur 12 tirages possibles. La probabilité de tirer un double est donc égale à $\frac{3}{12}=\frac{1}{4}$.
3. (a) Il faut remplacer A par 1 000, B par 4 et C par 5.

   (b) Il faut insérer le bloc après répéter 1 000 fois.

   (c) Il faut insérer le bloc avant répéter 1 000 fois.

   (d) Il faut placer à la fin la proposition 2.

3 Nouvelle-Calédonie, février 2020, série générale, exercice 8

Références

Série : générale

Académie :Nouvelle-Calédonie

Date : février 2020

Exercice 8 - 10 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- frise
- déplacement
- nombres et calculs
- calcul sans programme
- grandeurs et mesures

Thèmes informatiques

- activité du lutin
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Nouvelle-Calédonie, Février 2020, exercice 8

On souhaite représenter 6 bassins rectangulaires à l'aide d'un logiciel de programmation comme sur la figure n° 1 ci-dessous :

La figure n'est pas à l'échelle

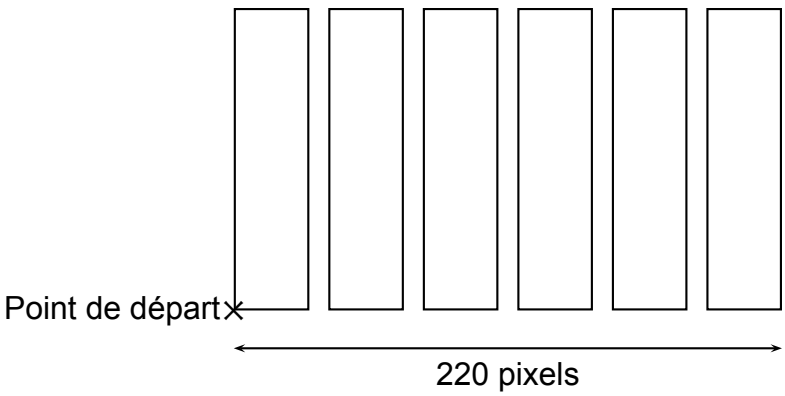
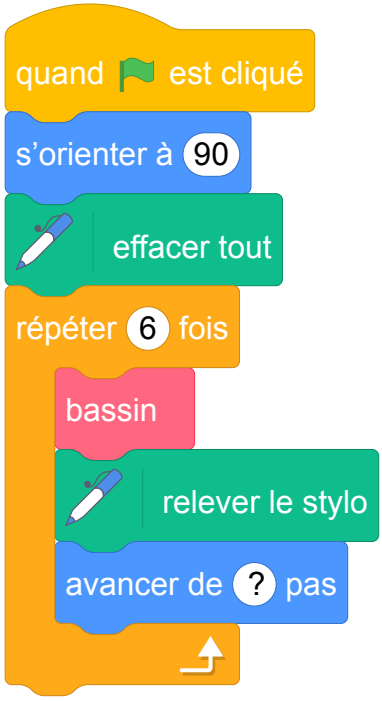


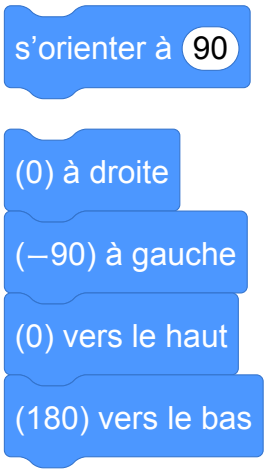
Figure n° 1

1. Compléter, en **annexe**, le script du bloc « bassin » pour qu'il permette de tracer un bassin rectangulaire de largeur 30 pas et de longueur 150 pas.

2. Le script ci-dessous doit permettre d'obtenir la figure n° 1. Il utilise le bloc bassin de l'**annexe**.



Rappel :



Sachant que la longueur totale de la figure n° 1 est de 220 pixels, quelle valeur doit être placée à la dernière ligne dans la consigne avancer de ? pas ? Justifier la réponse.

Dans cette question, toute trace de recherche, même incomplète ou non fructueuse, sera prise en compte dans l'évaluation.

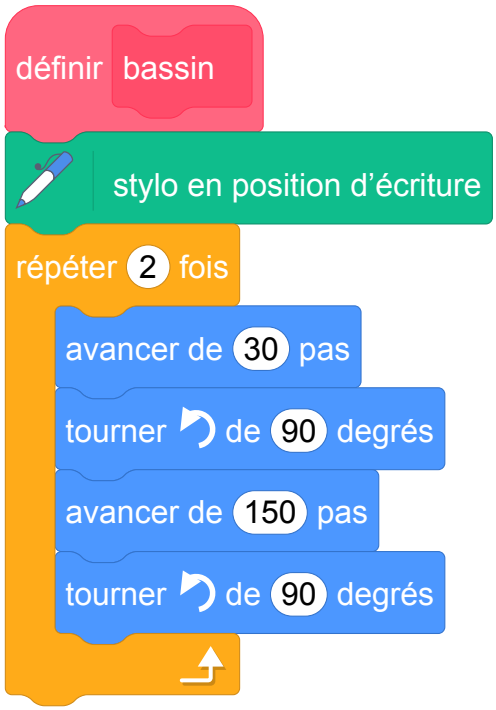
Annexe (à rendre avec la copie)

Correction

Fichier Scratch : `sg_nouvelle_caledonie_2020_fevrier_ex8.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/481466018/`

1. **Script à compléter**

Annexe (à rendre avec la copie)



Dans ce bloc il s'agit de tracer un rectangle. Il faut donc avancer d'une part de la longueur du rectangle et d'autre part de sa largeur. Les quatre angles d'un rectangle étant droits, il faut tourner de 90° à chaque fois.

2. avancer de 38 pas

La longueur d'une extrémité des bassins à l'autre est de 220 pas. Cette longueur correspond à 6 largeurs de bassin et 5 intervalles entre les bassins. Notons x un intervalle inter-bassins.
$5x + 6 \times 30 = 220$
$5x + 180 = 220$
$5x = 220 - 180$
$5x = 40$
$x = 40 \div 5$
$x = 8$ pas
A chaque étape le lutin doit avancer de la largeur du rectangle plus l'intervalle, soit de $30 + 8 = 38$ pas.

Analyse

Les explications de la commande s'orienter à 90 ne doivent pas être écrit sous forme d'instruction Scratch.

(0) à droite
(−90) à gauche
(0) vers le haut
(180) vers le bas

4 Polynésie, septembre 2020, série générale

Références

Série : générale

Académie : Polynésie

Date : septembre 2020

Exercice 2 - 15 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- repérage
- frise
- calcul sans programme
- grandeurs et mesures
- translation
- proportionnalité

Thèmes informatiques

- variable réservée
- activité du lutin
- calcul
- itération bornée
- bloc sans paramètre

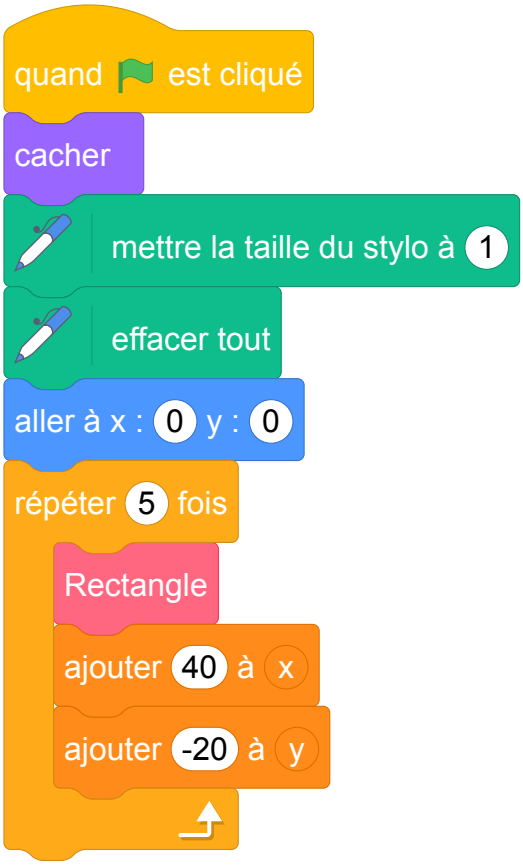
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, septembre 2020

On souhaite réaliser une frise composée de rectangles.
Pour cela, on a écrit le programme ci-dessous :



Script principal

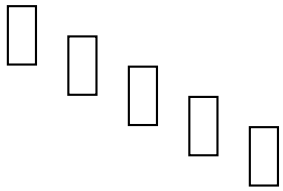
Bloc « rectangle »

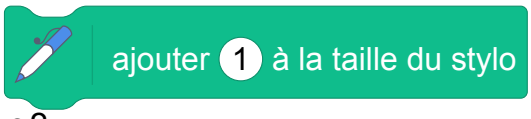
On rappelle que l'instruction « s'orienter à 90 » consiste à s'orienter horizontalement vers la droite.

Dans cet exercice, aucune justification n'est demandée.

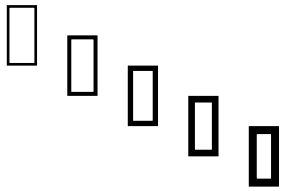
1. Quelles sont les coordonnées du point de départ du tracé ?
2. Combien de rectangles sont dessinés par le script principal ?
3. Dessiner à main levée la figure obtenue avec le script principal.

4. (a) Sans modifier le script principal, on a obtenu la figure ci-dessous composée de rectangles de longueur 40 pas et de largeur 20 pas. Proposer une modification du bloc « rectangle »permettant d'obtenir cette figure.



(b) Où peut-on alors ajouter l'instruction 

dans le script principal pour obtenir la figure ci-dessous ?



Correction

Fichier Scratch : `sg_polynesie_2020_septembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/481463960/`

1. Le point de départ a pour coordonnées (0 ; 0).
2. Cinq rectangles sont dessinés comme l'indique le paramètre de la boucle.
3. On obtient un rectangle de longueur 40 et de largeur 20.
4. (a) Il suffit d'échanger le 40 et le 20 des instructions « avancer » dans le bloc « Rectangle ».
   (b) Il faut ajouter cette instruction dans le bloc « répéter 5 fois » à la fin de la séquence d'instructions.

5 Métropole, septembre 2020, série générale

Références

Série : générale

Académie : Métropole-La Réunion

Date : Septembre 2020

Exercice 5 - 14 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- rosace
- déplacement
- nombres et calculs
- calcul sans programme
- grandeurs et mesures
- rotation

Thèmes informatiques

- activité du lutin
- itération bornee
- bloc sans paramètre

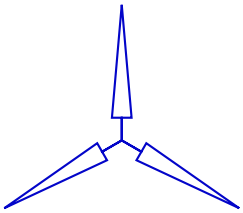
Compétences travaillées

- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

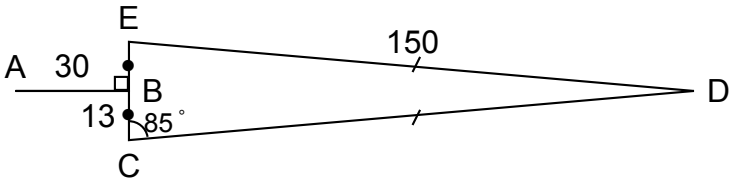
Énoncé - Métropole, septembre 2020

On cherche à dessiner une éolienne avec le logiciel Scratch ; elle est formée de 3 pales qui tournent autour d'un axe central.



Éolienne modélisée par Scratch

1. La figure ci-dessous représente une pale d'éolienne.



- DEC est un triangle isocèle en D ;
- B est le milieu de [EC] ;
- [AB] est perpendiculaire à [EC] ;
- $\widehat{ECD} = 85^\circ$.

a. Montrer que l'angle $\widehat{CDE} = 10^\circ$.

b. Le script pale ci-contre permet de tracer une pale de l'éolienne avec le logiciel Scratch.
Pourquoi la valeur indiquée dans le bloc de la ligne n° 6 est-elle 95 ?

c. Dans ce même script pale, par quelle valeur doit-on compléter le bloc situé à la ligne n° 8 ?
Recopier cette valeur sur votre copie.

2. Le script éolienne ci-contre permet de tracer l'éolienne avec le logiciel Scratch.
Par quelle valeur doit-on compléter la boucle « répéter » ?
Recopier cette valeur sur votre copie.

Correction

Fichier Scratch : `sg_metropole_2020_septembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/481464187/`

1. (a) Les angles à la base du triangle EDC isocèle en D, ont la même mesure.

   On sait que la somme des mesures des trois angles est égale à 180 en degrés. Donc $85 + 85 + \widehat{EDC} = 180$, d'où $\widehat{EDC} = 180 - 170 = 10^\circ$.

   (b) Après la ligne 5, on est en C dans la direction opposée à celle de E ; pour aller vers D il faut tourner vers la gauche du supplémentaire de l'angle de mesure 85°, soit de $180 - 85 = 95^\circ$.

   (c) Après la ligne 9, on est en D, dans la direction opposée de celle de C. Pour aller vers E, il faut faire demi-tour donc tourner vers la gauche de 180° et de revenir de 10°, donc de tourner vers la gauche de 170°.

2. Il y a 3 pales : il faut donc répéter 3 fois le script « pale ».

6 Nouvelle-Calédonie, décembre 2020, série générale

Références

Série : générale

Académie :Nouvelle-Calédonie

Date : décembre 2020

Exercice 8 - 11 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- caractérisation des figures
- spirale
- nombres et calculs
- calcul sans programme
- grandeurs et mesures

Thèmes informatiques

- variable déclarée
- activité du lutin
- itération bornée

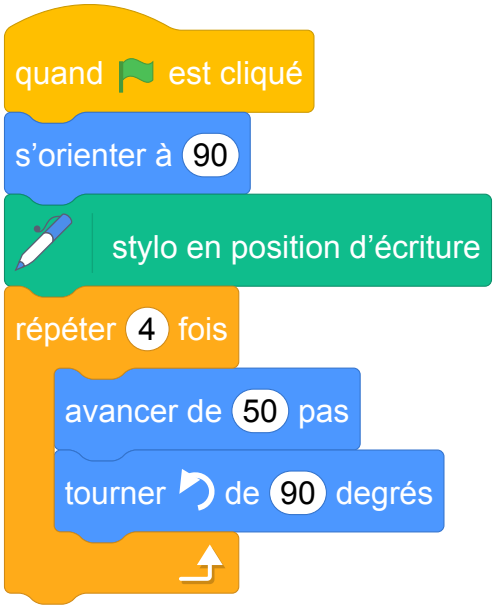
Compétences travaillées

- comprendre un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Nouvelle-Calédonie, Décembre 2020

Le script suivant permet de tracer le carré de côté 50 unités .



1. Sur l'annexe , compléter le script pour obtenir un triangle équilatéral de coté 80 unités.
2. On a lancé le script suivant :

Entourer sur l'annexe , la figure obtenue avec ce script.

ANNEXE

1. **Script à compléter**

2.

Figure 1

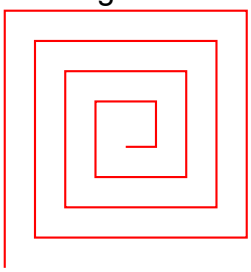


Figure 2

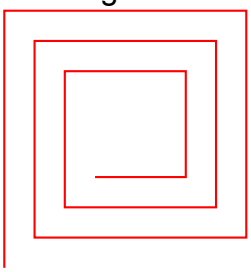
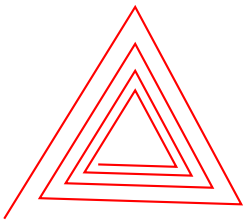


Figure 3

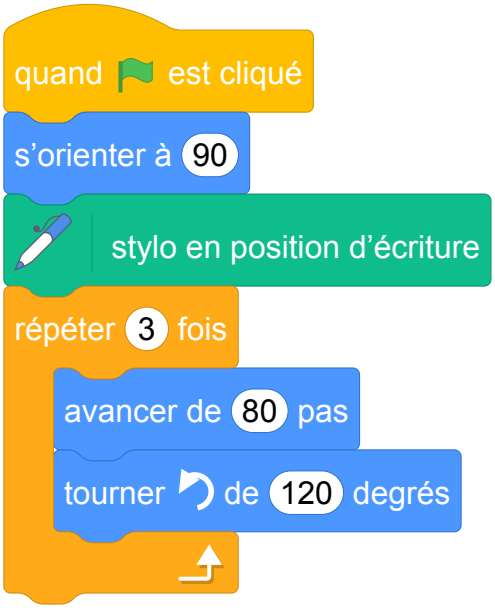


Correction

Fichier Scratch : `sg_nouvelle_caledonie_2020_decembre.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/481464209/`

1. **Script à compléter**



2. Cela ne peut pas être la figure 3 car le lutin tourne de 90° et non de 120°. Dans le script, la longueur est initialitsée à 40 et augmente de 10 à chaque étape. Cela ne peut donc pas être la figure 1 car entre les deux premiers segments, le rapport est de 2.

Figure 1

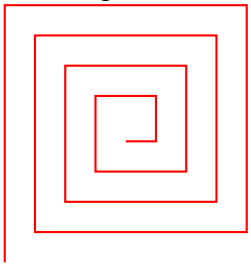


Figure 2

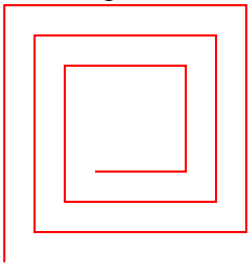
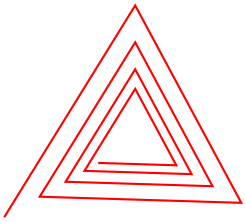


Figure 3



Brevets 2021

1 Asie, juin 2021, série générale

Références

Série : générale

Académie : Asie

Date : juin 2021

Exercice 3 - 21 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- calcul sans programme

Thèmes informatiques

- variable réservée
- calcul
- branchement conditionnel

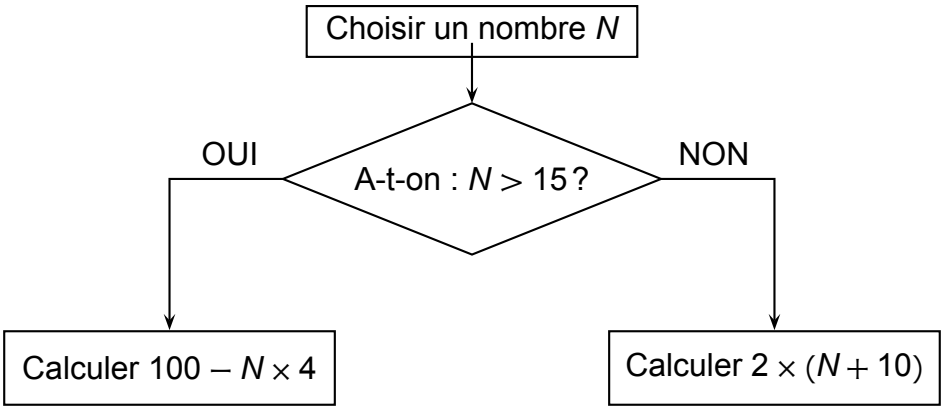
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

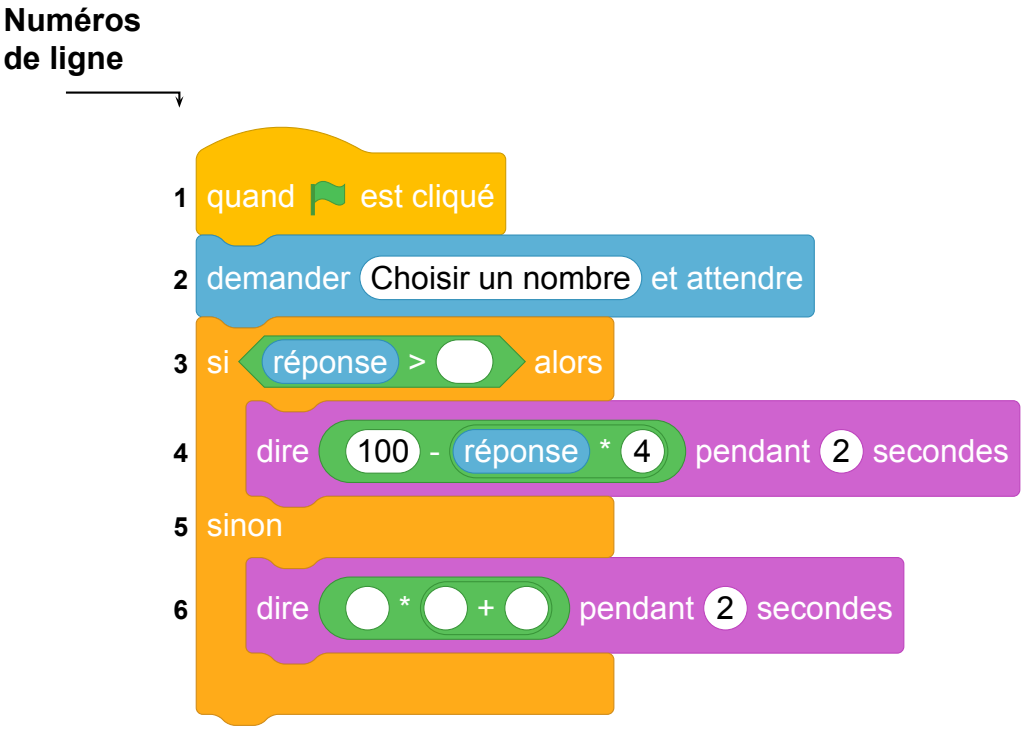
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Asie, 21 juin 2021

Voici un algorithme :



1. Justifier que si on choisit le nombre N de départ égal à 18, le résultat final de cet algorithme est 28.
2. Quel résultat final obtient-on si on choisit 14 comme nombre N de départ ?
3. En appliquant cet algorithme, deux nombres de départ différents permettent d'obtenir 32 comme résultat final. Quels sont ces deux nombres ?
4. On programme l'algorithme précédent :



   (a) Recopier la ligne 3 en complétant les pointillés :
   ligne 3 : si réponse $>$ ... alors

   (b) Recopier la ligne 6 en complétant les pointillés :
   ligne 6 : dire ... $*$ (... $+$ ...) pendant 2 secondes

5. On choisit au hasard un nombre premier entre 10 et 25 comme nombre N de départ.
   Quelle est la probabilité que l'algorithme renvoie un multiple de 4 comme résultat final ?

Correction

1. $18 \longrightarrow 18 > 10 \longrightarrow 100 - 18 \times 4 = 100 - 72 = 28$
2. $14 \longrightarrow 14 < 10 \longrightarrow 2 \times (14 + 10) = 2 \times 24 = 48$. On obtient 48.
3. Appelons x le premier nombre et supposons que $x > 15$. On a :

$$\begin{aligned} 100 - x \times 4 &= 32 \\ 100 - x \times 4 - 100 &= 32 - 100 \\ -x \times 4 &= -68 \\ \frac{-x \times 4}{-4} &= \frac{-68}{-4} \\ x &= 17 \end{aligned}$$

On vérifie que $17 > 15$ c'est le seul nombre strictement supérieur à quinze solution du problème.

Appelons y le deuxième nombre et supposons que $y \leq 15$. On a :

$$\begin{aligned} 2 \times (y + 10) &= 32 \\ \frac{2 \times (y + 10)}{2} &= \frac{32}{2} \\ y + 10 &= 16 \\ y + 10 - 10 &= 16 - 10 \\ y &= 6 \end{aligned}$$

On vérifie que $6 \leq 15$. C'est le seul nombre inférieur ou égal à quinze solution du problème.

Les deux nombres solutions sont donc 6 et 17.

4. (a) Si réponse >15 alors...
   (b) dire 2 *(réponse+10)pendant 2 secondes
5. Les nombres premiers compris entre 10 et 25 sont :
   - $11 < 15$ dont l'image par l'algorithme est : $2 \times (11 + 10) = 42 = 4 \times 10,5$
   - $13 < 15$ dont l'image par l'algorithme est : $2 \times (13 + 10) = 46 = 4 \times 11,5$
   - $17 > 15$ dont l'image par l'algorithme est : $100 - 4 \times 17 = 4 \times 25 - 4 \times 17 = 4 \times (25 - 17)$
   - $23 > 15$ dont l'image par l'algorithme est : $100 - 4 \times 23 = 4 \times (25 - 23)$

   Il y a donc quatre issues dont deux sont des multiples de quatre. Il y a équiprobabilité donc la probabilité d'obtenir un multiple de quatre est : $\frac{2}{4}\left(= \frac{1}{2}\right)$

Analyse

Commentaires de l'IREM de Marseille :

— le logigramme est métaphorique. Peut-on vraiment parler d'algorithme ?
— Comment justifier le résultat d'un calcul ?
— On peut questionner l'utilisation d'un texte à trous.
— La formulation des questions 1, 2 et 4(a) à revoir.
— Remarque : dans les prg de calcul en maths il n'y a jamais de sortie (pas de "afficher le résultat")

2 Centres étrangers, juin 2021, série générale

Références

Série : générale générale

Académie : Centres étrangers

Date : juin 2021

Exercice 3 - 16 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable réservée
- variable déclarée
- calcul

Compétences travaillées

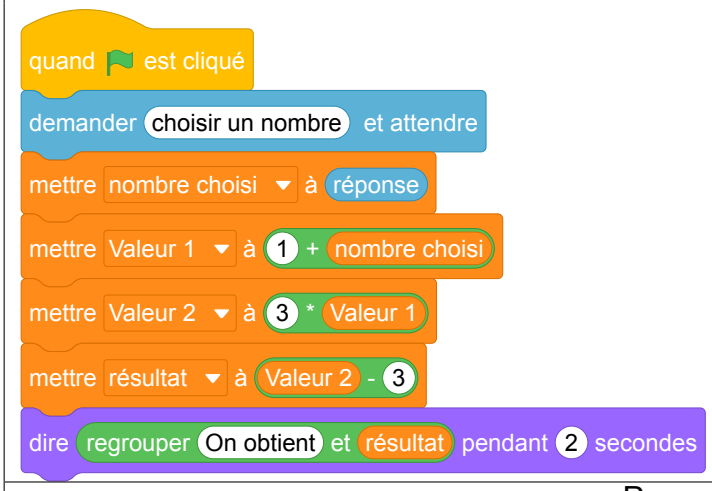
- Aucune compétence algorithmique

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

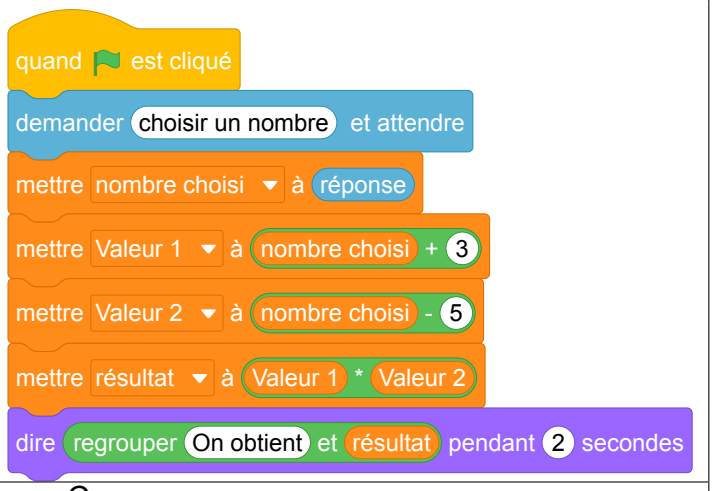
Énoncé - Centres étrangers, 15 Juin 2021

Un professeur propose à ses élèves trois programmes de calculs, dont deux sont réalisés avec un logiciel de programmation.

Programme A



Programme B



Programme C

- Choisir un nombre
- Multiplier par 7
- Ajouter 3
- Soustraire le nombre de départ

1. (a) Montrer que si on choisit 1 comme nombre de départ alors le programme A affiche pendant 2 secondes « On obtient 3 ».

   (b) Montrer que si on choisit 2 comme nombre de départ alors le programme B affiche pendant 2 secondes « On obtient -15 ».

2. Soit x le nombre de départ, quelle expression littérale obtient-on à la fin de l'exécution du programme C ?

3. Un élève affirme qu'avec un des trois programmes on obtient toujours le triple du nombre choisi. A-t-il raison ?

4. (a) Résoudre l'équation $(x+3)(x-5)=0$.

   (b) Pour quelles valeurs de départ le programme B affiche-t-il « On obtient 0 » ?

5. Pour quelle(s) valeur(s) de départ le programme C affiche-t-il le même résultat que le programme A ?

Correction

Fichier scratch : `sg_centres_etrangers_2021_juin.sb3`

Lien vers le fichier Scratch en ligne : `scratch.mit.edu/projects/577772006/`

1. (a) On obtient successivement : $1 \to 1+1=2 \to 3\times 2=6 \to 6-3=3$.

   (b) On obtient successivement : $2 \to 2+3=5 \to 2-5=-3 \to 5\times -3=-15$.

2. Soit x le nombre de départ, quelle expression littérale obtient-on à la fin de l'exécution du programme C ? On obtient successivement : $x \to x\times 7 \to 7x+3 \to 7x+3-x=6x+3$.

3. On vient de voir que le programme C donne $6x+3 \neq 3x$;

   Le programme A donne à partir de x : $x \to 1+x \to 3(1+x)=3+3x \to 3+3x-3=3x$: on obtient bien le triple.

   Le programme B donne à partir de x :

   $x \to x+3 \to x-5 \to (x+3)(x-5)=x^2-5x+3x-15=x^2-2x-15 \neq 3x$.

   L'élève a raison.

4. (a) U produit de deux facteurs est nul si l'un des facteurs est nul, donc :

   $(x+3)(x-5)=0$ si $\left\{\begin{array}{rcl} x+3 & = & 0 \\ x-5 & = & 0 \end{array}\right.$ ou encore $\left\{\begin{array}{rcl} x & = & -3 \\ x & = & 5 \end{array}\right.$

   L'ensemble des solutions est $S=\{-3\ ;\ 5\}$.

   (b) On a vu que le programme B donne à partir de x le produit $(x+3)(x-5)$ ry on a vu dans la question précédente que -3 et 5 annulaient ce produit.

   Donc le programme B donne à partir de -3 et à partir de 5 le nombre 0.

5. Il faut trouver x tel que $6x+3=3x$ soit en ajoutant à chaque membre $-3x$: $3x+3=0$ ou $3x=-3$, soit $3\times x=3\times(-1)$ et finalement $x=-1$

   Le nombre -1 donne par A ou C le même résultat -3.

Analyse

Comme pour la plupart des énoncés traitant de calcul littéral, l'utilisation de Scratch est complètement inutile. Les questions ne permettent pas d'évaluer les compétences en algorithmique et programmation. Ce programme de calcul contient encre un nombre délirant de variables.

On note une formulation. inutilement complexe de la question 1 (a),

3 Métropole, La Réunion, Antilles-Guyanne, juin 2021, série générale

Références

Série : générale

Académie : Métropole, La Réunion, Antilles, Guyanne

Date : juin 2021

Exercice 4 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable réservée
- variable déclarée
- calcul

Compétences travaillées

- exécuter un algorithme
- compléter un algorithme

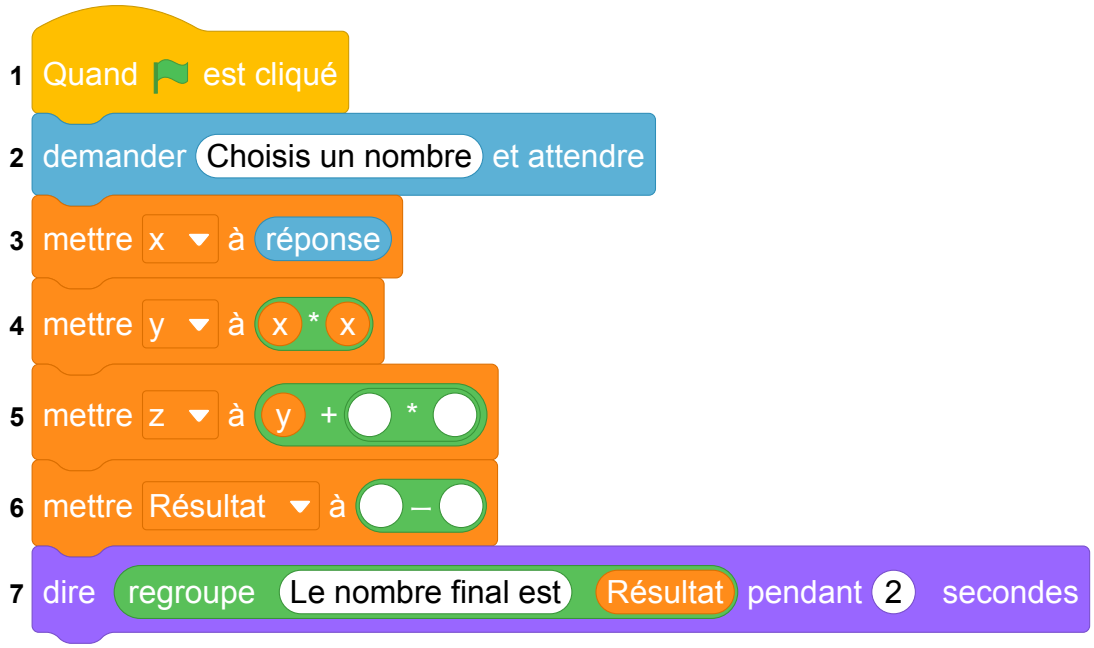
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion – Antilles-Guyane, 28 juin 2021

Voici un programme de calcul :

Choisir un nombre. Prendre le carré du nombre de départ. Ajouter le triple du nombre de départ. Soustraire 10 au résultat.

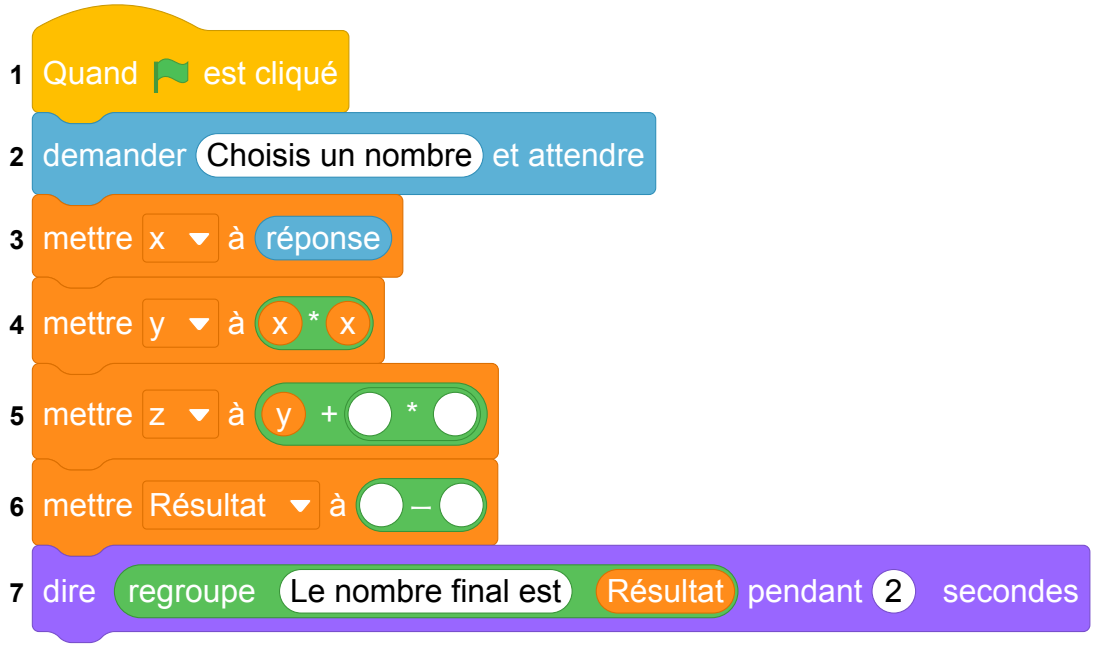
1. Vérifier que si on choisit 4 comme nombre de départ, on obtient 18
2. Appliquer ce programme de calcul au nombre - 3
3. Vous trouverez ci-dessous un script, écrit avec scratch.



Compléter sur l'ANNEXE les lignes 5 et 6 pour que ce script corresponde au programme de calcul.

4. On veut déterminer le nombre à choisir au départ pour obtenir zéro comme résultat.
   (a) On appelle x le nombre de départ. Exprimer en fonction de x le résultat final.
   (b) Vérifier que ce résultat peut aussi s'écrire sous la forme $(x+5)(x-2)$.
   (c) Quel(s) nombre(s) doit-on choisir au départ pour obtenir le nombre 0 à l'arrivée ?

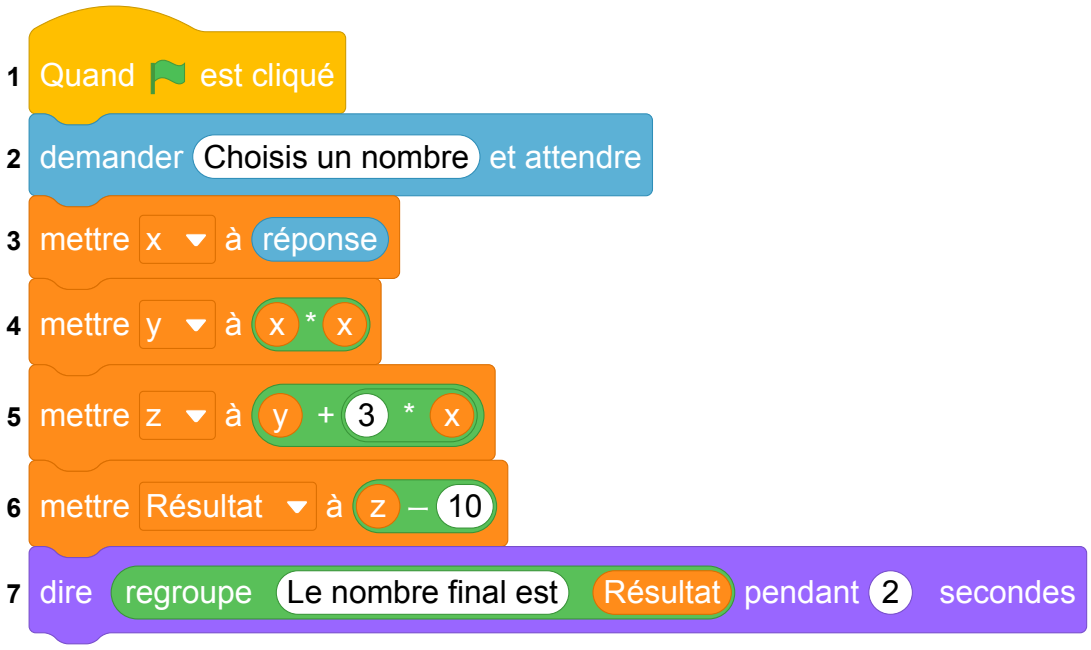
ANNEXE



Correction

Fichier scratch : `sg_metropole_reunion_antilles_guyanne_2021_juin.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/577789132/`

1. $4^2 + 3 \times 4 - 10 = 16 + 12 - 10 = 18$
2. $(-3)^2 + 3 \times (-3) - 10 = 9 - 9 - 10 = -10$.
3. On complète l'annexe scratch comme ci-dessous.



4. (a) On note x le nombre initial.
   $y = x^2$
   $z = y + 3x = x^2 + 3x$
   $resultat = z - 10 = x^2 + 3x - 10 =$
   Le résultat final est : $x^2 + 3x - 10$.
   (b) On développe : $(x+5)(x-2) = x^2 - 2x + 5x - 5 \times 2 = x^2 + 3x - 10$.
   (c) Le résultat final est nul si et seulement si : $(x+5)(x-2) = 0$.
   Il s'agit d'un produit nul. Cela équivaut donc à : $x + 5 = 0$ ou $x - 2 = 0$.
   Soit $x = -5$ ou $x = 2$.
   Pour obtenir 0 à la fin, il n'y a que deux nombres possibles au départ : -5 ou 2.

Analyse

Commentaires de l'IREM de Marseille :

Encore une fois cet exercice n'a pas vraiment de lien avec le cursus d'algorithmique et programmation. C'est encore un programme avec un nombre important de variables qui paraphrase un programme de calcul mathématiques.

4 Polynésie, juin 2021, série générale

Références

Série : générale

Académie : Polynésie

Date : juin 2021

Exercice 4 - 19 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- agrandissement réduction
- caractérisation des figures
- frise
- nombres et calculs
- calcul sans programme

Thèmes informatiques

- variable déclarée
- activité du lutin
- calcul
- itération bornée
- bloc sans paramètres

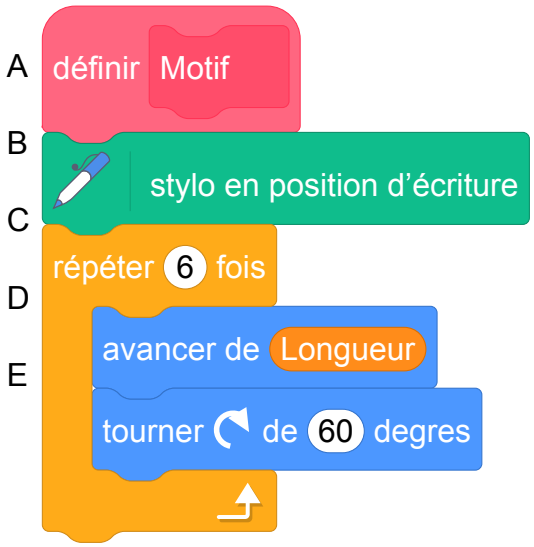
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, 25 juin 2021

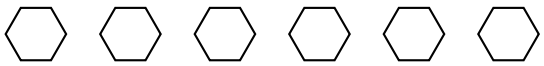
On donne le programme suivant :



On rappelle que « s'orienter à 90 » signifie que l'on est orienté vers la droite.

1. On prendra dans cette question 1 mm pour un pixel.
   Représenter en vraie grandeur sur votre copie la figure que trace le bloc Motif lorsque Longueur vaut 30 pixels.
2. Ce programme utilise une variable, quel est son nom ? À quoi correspond-elle sur la figure réalisée par le bloc Motif ?
3. Laquelle de ces trois figures obtient-on lorsqu'on exécute ce programme ? Indiquer sur la copie le numéro de la bonne proposition parmi les trois suivantes. On expliquera son choix.

4. Modifier le programme précédent pour obtenir la figure ci-dessous. Pour cela, indiquer les numéros des instructions à supprimer ou à modifier, et préciser les modifications à apporter :



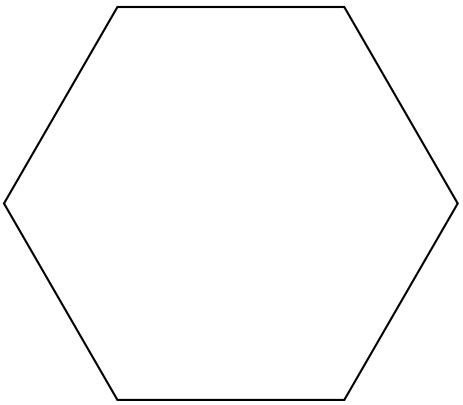
5. On souhaite modifier le bloc Motif afin qu'il permette de tracer un carré. Pour cela, indiquer les lettres des instructions à supprimer ou à modifier, et préciser les modifications à apporter.

Correction

Fichier scratch : `sg_polynesie_2021_juin.sb3`

Lien vers le fichier Scratch en ligne : `scratch.mit.edu/projects/577787151/`

1.



2. La variable est « Longueur » qui correspond à la longueur du côté de l'hexagone tracé par par le bloc Motif.
3. On dessine quatre hexagones après s'être déplacé vers la droite en augmentant à chaque fois la longueur du côté : c'est donc la figure 2 qui est produite.
4. Il faut :
   — garder la taille de l'hexagone dessiné par le Motif en supprimant la ligne 9.
   — modifier le nombre de répétitions ligne (5) en "répéter 6 fois" au lieu de 4.
5. Pour obtenir un carré il y a deux lignes à modifier. Il faut :
   — répéter 4 fois (ligne C),
   — tourner de 90° (ligne E).

Analyse

Commentaires de l'IREM de Marseille :

Il s'agit de produire des constructions géométriques à partir d'un programme Scratch. L'exercice des plutot bien construit car il permet d'évaluer des compétences en algorithmique.

On peut envisager d'adapter cet exercice en python pour l'utiliser en classe de seconde.

Propositions

Adaptation en python (turtle) possible pour reprise en classe de seconde.

5 Métropole, septembre 2021, série générale

Références

Série : générale

Académie : Métropole-La Réunion

Date : septembre 2021

Exercice 4 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie sans transformation
- caractérisation des figures
- déplacement
- grandeurs et mesures
- périmètre

Thèmes informatiques

- activité du lutin
- itération bornée

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

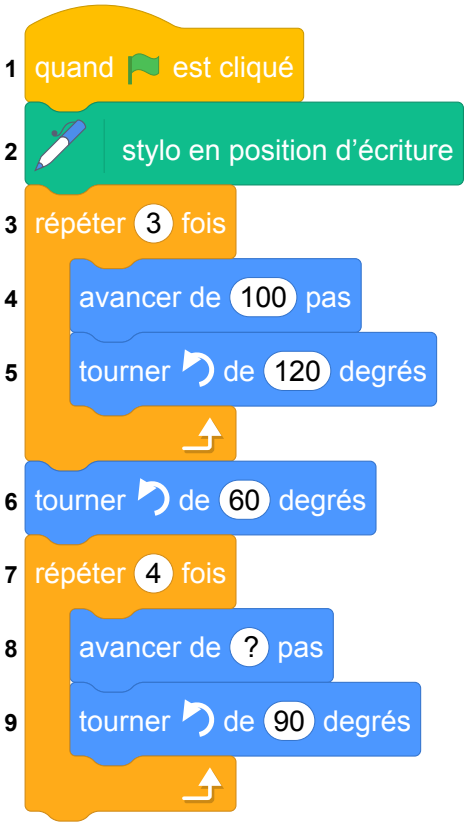
Énoncé - Métropole, 13 septembre 2021

Voici trois programmes réalisés avec l'application Scratch.

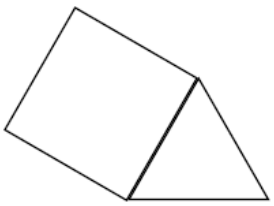
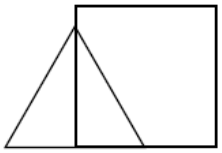
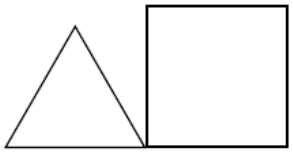
Programme 1

Programme 2

Programme 3



1. Ils donnent les trois figures suivantes constituées de triangles et de quadrilatères **identiques**.

Figure A	Figure B	Figure C
		

   (a) Quelle est la nature du triangle et du quadrilatère sur chaque figure ? Aucune justification n'est attendue.

   (b) Quelle est la valeur manquante à la ligne 8 dans ces 3 programmes ?

   (c) Indiquer sur la copie, pour chaque figure, le numéro du programme qui permet de l'obtenir.

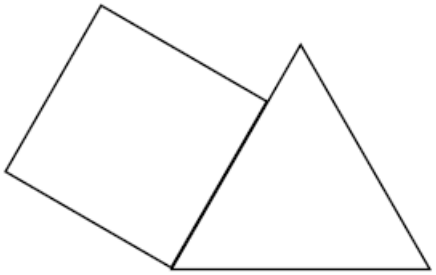
2. (a) Maintenant nous allons modifier les programmes précédents pour construire d'autres figures pour lesquelles le périmètre du quadrilatère est égal au périmètre du triangle. Quelle valeur du pas doit-on alors choisir à la ligne 8 de chaque programme ?

   (b) Représenter la figure A obtenue avec cette nouvelle valeur, en prenant 1 cm pour 25 pas.

Correction

Fichier Scratch : `sg_metropole_2021_septembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/577839207/`

1. (a) Dans chaque cas le triangle est équilatéral et le quadrilatère est un carré.
   (b) Avancer de 100 pas.
   (c) Programme 1 : figure B ; Programme 2 : figure C ; Programme 3 : figure A.
2. (a) Si c est la longueur du côté du carré et t la longueur du côté du triangle, on doit avoir $4c = 3t$.
   Donc si $t = 100$, alors $4c = 300$, soit $c = 75$.
   Il faut donc écrire à la ligne 8 : avancer de 75 pas.
   (b)



6 Amérique du Sud, novembre 2021, série générale

Références

Série : générale

Académie : Amérique du Sud

Date : novembre 2021

Exercice 4 - 14 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- pavage
- frise
- rosace
- grandeurs et mesures
- rotation
- proportionnalité

Thèmes informatiques

- scène
- activité du lutin
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

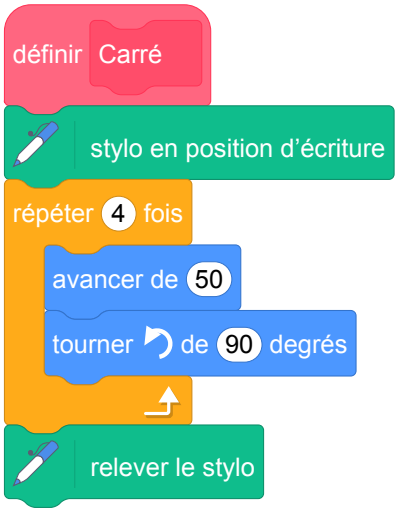
Énoncé - Amérique du Sud, novembre 2021

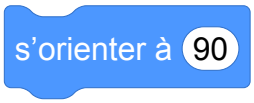
Dans tout cet exercice, aucune justification n'est demandée

On donne le programme suivant :

Script principal

le bloc Carré



On rappelle que l'instruction 

signifie que l'on s'oriente vers la droite.

1. On lance le programme.
   Construire la figure obtenue en prenant 1 cm pour 25 unités de longueur.
   On modifie le Script principal et on obtient deux scripts ci-dessous :

Script principal A

Script principal B

2. Parmi les trois figures ci-dessous, associer sur votre copie chacun des deux scripts principaux A et B à la figure qu'il permet de réaliser :

Figure 1

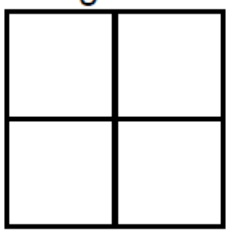
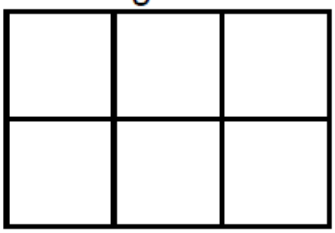
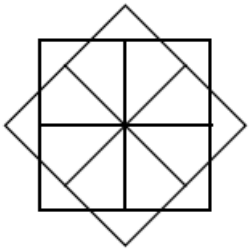


Figure 2

Figure 3

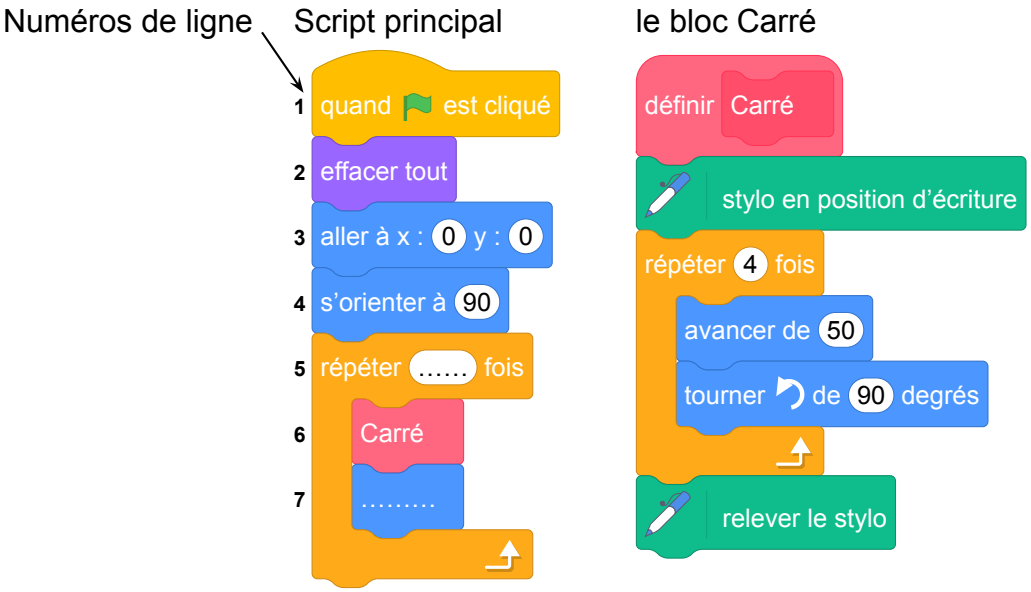


On souhaite réaliser la figure suivante :



Le point de départ se situe au centre de la figure.

3. Compléter le nouveau script principal ci-dessous en recopiant sur la copie uniquement les lignes 5 et 7. Pour mémoire, l'énoncé rappelle ci-dessous à droite le descriptif du bloc Carré.



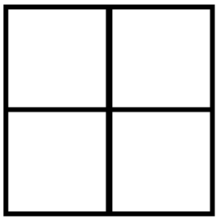
Correction

Fichier Scratch : `sg_amerique_du_sud_2021_novembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/638036598/`

1. On lance le programme et on construit la figure obtenue (1 cm pour 25 unités de longueur).

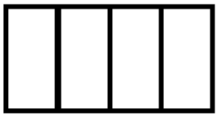
2.

Figure 1



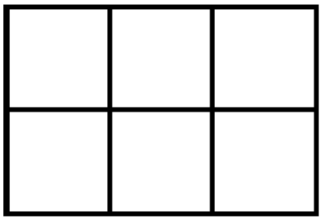
Script principal B

Figure 2

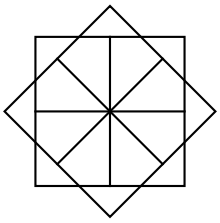


Script principal A

Figure 3



On souhaite réaliser la figure suivante :



Le point de départ se situe au centre de la figure.

3. On donne le nouveau script principal ci-dessous.

Numéros de ligne — Script principal

1 quand est cliqué
2 effacer tout
3 aller à x : 0 y : 0
4 s'orienter à 90
5 répéter 8 fois
6 Carré
7 tourner de 45 degrés

le bloc Carré

définir Carré
stylo en position d'écriture
répéter 4 fois
avancer de 50
tourner de 90 degrés
relever le stylo

On complète les lignes 5 et 7.

— Ligne 5 : « répéter 8 fois » (il y a 8 carrés tracés)

— Ligne 7 : on passe d'un carré au suivant en effectuant une rotation d'angle 45° dont le centre est le centre de la figure donc on utilise tourner de 45 degrés ou tourner de 45 degrés

Propositions

En classe, il pourrait être intéressant de demander aux élèves de modifier le script principal pour obtenir la figure 3.

7 Amérique du Nord, juin 2021, série générale

Références

Série : générale

Académie : Amérique du Nord

Date : juin 2021

Exercice 4 - 16 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

Thèmes informatiques

- état

Compétences travaillées

- exécuter un algorithme
- écrire un algorithme

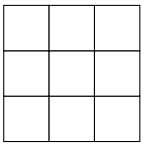
Compétences travaillées

- exécuter un algorithme
- écrire un algorithme

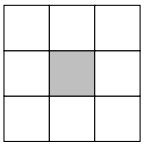
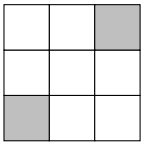
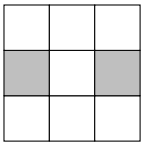
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	1

Énoncé - Amérique du Nord, 3 juin 2021

Dans cet exercice, aucune justification n'est demandée.
On dispose d'un tableau carré ci-contre partagé en neuf cases blanches de mêmes dimensions qui constituent un motif.



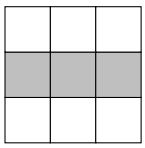
Quatre instructions A, B, C et E permettent de changer l'aspect de certaines cases, lorsqu'on applique ces instructions. Ainsi :

Instruction	Descriptif	Effet de l'instruction
A	La case centrale du motif est noircie.	
B	Dans le motif, la case en bas à gauche et la case en haut à droite sont noircies.	
C	Dans le motif, la case médiane à gauche et la case médiane à droite sont noircies.	
E	Les couleurs du motif sont inversées : les cases blanches deviennent noires et les les cases noires deviennent blanches.	Inverser les couleurs

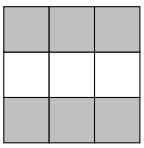
Remarque : si une case du motif est déjà noire et une instruction demande de la noircir, alors cette case ne change pas de couleur et reste noire à la suite de cette instruction.

Exemples : à partir d'un motif dont toutes les cases sont blanches :

la suite d'instructions A C permet d'obtenir ce motif



la suite d'instructions A C E permet d'obtenir ce motif

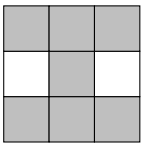


Pour chacune des questions suivantes, on dispose au départ d'un motif dont toutes les cases sont blanches.

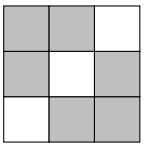
1. Représenter le motif obtenu avec la suite d'instructions A B.
2. Parmi les quatre propositions suivantes, deux propositions permettent d'obtenir le motif ci-contre. Lesquelles ?

   Proposition n° 1 : A B C **Proposition n° 3 :** B C E C

   Proposition n° 2 : C E **Proposition n° 4 :** C A E A

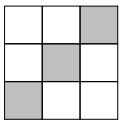


3. Donner une suite d'instructions qui permet d'obtenir le motif ci-contre.



Correction

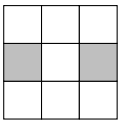
1. Le motif obtenu avec la suite d'instructions A B est :



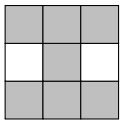
2. Les propositions 2 et 4 permettent d'obtenir le motif demandé.

Proposition 2 : C E

En appliquant l'instruction C on obtient :

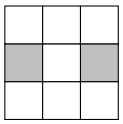


Puis en inversant les couleurs avec l'instruction E, on obtient :

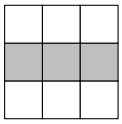


Proposition 4 : C A E A

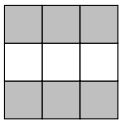
En appliquant l'instruction C on obtient :



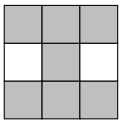
Puis en appliquant l'instruction A on obtient :



L'instruction E inverse les couleurs on obtient donc :

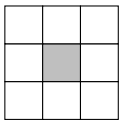


L'instruction A va changer la couleur de la case centrale :

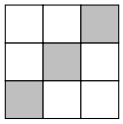


3. La suite A B E permet d'obtenir la diagonale montante blanche.

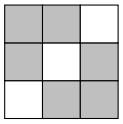
En appliquant l'instruction A, on obtient :



Puis, en appliquant l'instruction B, on obtient :



En inversant les couleurs avec l'instruction A on a :



Analyse

Le sujet est original parce qu'il ne comporte pas de Scratch et porte sur la compétence au programme « écrire, mettre au point (tester, corriger) et exécuter un programme en réponse à un problème donné ». C'est un exercice de nature algorithmique qui permet de travailler la notion de changement d'état avec une situation où celui-ci est complètement visible.

Ces suites d'instructions ne sont pas exactement des programmes mais de « simples » automates. La situation (état) initiale est unique.

Il y a trop d'implicite en début de sujet. La notion d'état des cases du tableau n'est pas claire. Il n'est pas précisé qu'il y a deux couleurs donc deux états possibles pour chaque case : couleur blanche ou grise.

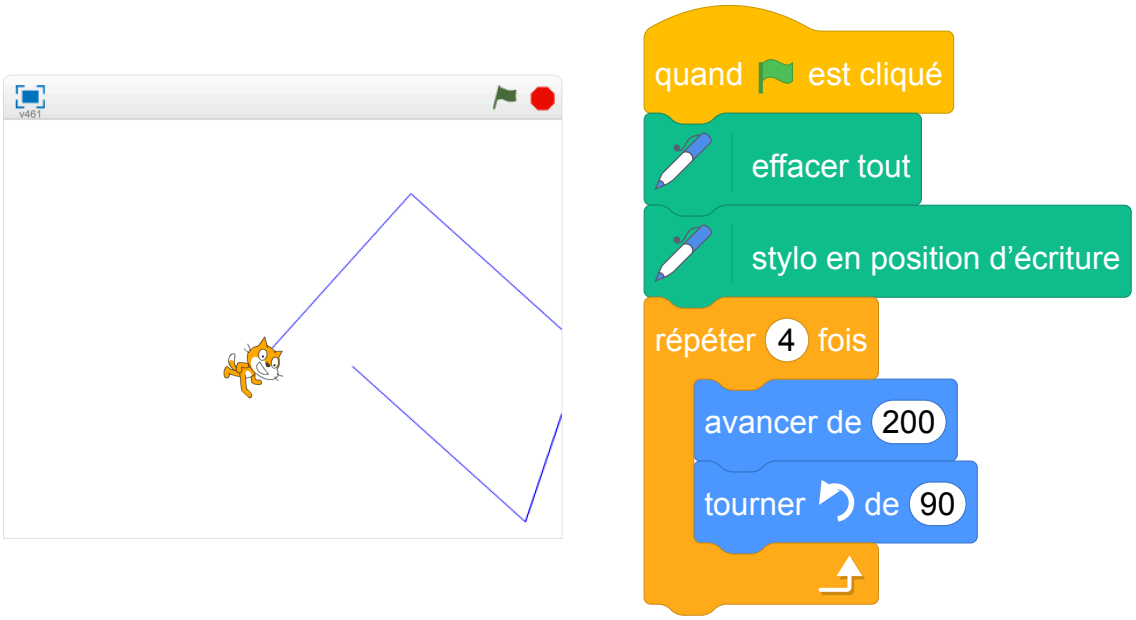
La question 1 permet de s'approprier la situation. La question 3 est très facile si la 1 est réussie car il s'agit de changer l'état de toutes les cases à partir de sa solution avec l'instruction E. La question 2 est très simple avec l'exemple donné. Il pourrait être intéressant de modifier le 2e exemple afin de rendre la question 2 moins triviale.

Propositions

Sensibilisation à la notion d'état

L'environnement d'exécution d'un programme Scratch est très riche. L'ensemble de tous les paramètres de la scène, la présence, la position et l'orientation de chaque lutin, les réglages de chaque stylo, etc. sont autant d'éléments qui constituent ce que l'on nomme généralement *l'état* du système. L'état initial du système au moment où l'utilisateur clique sur le bouton « démarrer » détermine très souvent l'issue observable du programme. C'est donc un aspect essentiel du travail.

Un grand nombre de blocs de Scratch ont un effet perceptible sur l'état. En informatique, on parle d'*effets de bord*. Par exemple, après l'exécution d'une instruction avancer de 200, le lutin actif a en général changé de position (même s'il est invisible), et un segment a peut-être été tracé sur la scène. Dans certains cas, si par exemple le lutin rencontre un bord de la scène avant d'avoir effectué son déplacement complet, sa position finale n'est en réalité pas celle à laquelle le programmeur peut s'attendre, ce qui peut avoir des effets surprenants par la suite. Voici un exemple d'issue observée après l'exécution d'un script destiné à tracer un carré :



Spécifier l'effet d'un bloc sur l'état. Dans les trois énoncés rencontrés dans cette section, des blocs personnalisés (ou *procédures*) sont utilisés pour décrire des tracés de figures géométriques, l'objectif final étant d'utiliser ces blocs dans des boucles afin de tracer des rosaces ou des frises. On est ici confronté à la notion de la *spécification* des effets d'un bloc sur l'environnement, ou de *contrat* que le bloc remplit. Ici, les deux blocs traçant des polygones (Losange, Motif) respectent au moins les deux conditions suivantes :

1. le tracé de la figure prend en compte la position et la direction du lutin au moment où le bloc est appelé ;
2. à la fin de l'exécution du bloc, le lutin est revenu à la même position (et orienté dans la même direction) qu'au début de l'exécution du bloc.

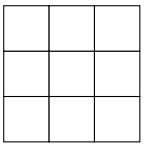
Ainsi, on pourra obtenir les effets voulus en faisant varier l'état entre deux appels successifs (par exemple on obtient une rosace en faisant varier la direction du lutin, ou une frise en faisant varier sa position). Par contre, le bloc maison utilisé dans l'énoncé 3.1.1 ne préserve pas la position du lutin. Il est indispensable de prendre cet aspect en compte dans la conception du script principal.

Une sensibilisation des élèves à ces problématiques liées à l'état de l'environnement d'exécution, et aux « contrats » qu'un programmeur associe aux éléments de ses programmes, nous parait fondamentale. À cette fin, l'utilisation de *commentaires* (permise par Scratch) pourrait se révéler utile.

Modification de l'énoncé

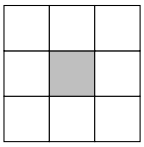
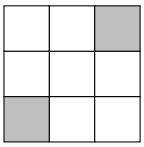
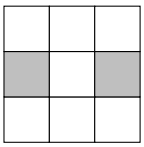
Dans cet exercice, aucune justification n'est demandée.

On dispose du tableau carré ci-contre partagé en neuf cases. Chacune de ces cases est colorée en blanc ou en gris. Le tableau constitue alors un motif. Quatre instructions A, B, C, E décrites ci-dessous permettent de changer la couleur des cases. Pour chaque question de l'exercice on se placera dans une situation initiale où toutes les cases sont blanches.



Dans les exemples ci-dessous, « Effet de l'instruction » indique le résultat de l'instruction sur la grille initiale vierge.

Ainsi :

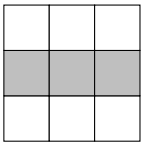
Instruction	Descriptif	Effet de l'instruction
A	La case centrale du motif est grisée.	
B	Dans le motif, la case en bas à gauche et la case en haut à droite sont grisées.	
C	Dans le motif, la case médiane à gauche et la case médiane à droite sont grisées.	
E	Les couleurs du motif sont inversées : les cases blanches deviennent grises et les les cases grises deviennent blanches.	Inverser les couleurs

Remarque : si une case du motif est déjà grise et une instruction demande de la griser, alors cette case ne change pas de couleur et reste grise.

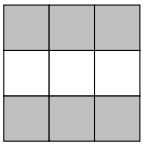
Exemples :

— à partir d'un motif dont toutes les cases sont blanches :

la suite d'instructions A C permet d'obtenir ce motif

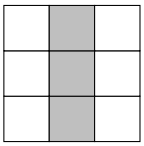


la suite d'instructions A C E permet d'obtenir ce motif

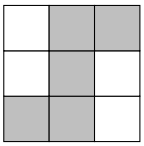


— à partir du motif suivant

motif initial



la suite d'instructions A B permet d'obtenir ce motif

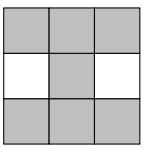


Pour chacune des questions suivantes, on dispose au départ d'un motif dont toutes les cases sont blanches.

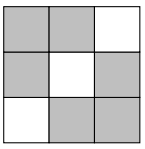
1. Représenter le motif obtenu avec la suite d'instructions A B.
2. Parmi les quatre propositions suivantes, deux propositions permettent d'obtenir le motif ci-contre. Lesquelles ?

   Proposition n° 1 : A B C **Proposition n° 3 :** B C E C

   Proposition n° 2 : C E **Proposition n° 4 :** C A E A



3. Donner une suite d'instructions qui permet d'obtenir le motif ci-contre.



Prolongements intéressants possibles

On peut s'intéresser aux aspects du problème relatifs à la notion d'automate et proposer les questions suivantes :

— Combien existe-t-il de grilles coloriés différentes ?

— Y a-t-il un motif que l'on ne peut pas obtenir avec cet automate ?

— Peut-on obtenir des motifs non symétriques ?

— Quelles sont les grilles coloriées que l'on peut obtenir à partir d'une grille blanche par application d'une séquence quelconque d'instructions ? Combien y en a-t-il ?

À un niveau NSI, on pourrait construire le graphe à la volée et rechercher le plus court chemin pour aller d'un état à un autre et réduire le problème à un parcours de graphe en largeur d'abord.

On peut également se poser la question d'un jeu d'instruction minimal pour que le graphe soit fortement connexe.

Un autre prolongement est de programmer le problème en Scratch ou avec une carte Microbit (https ://-makecode.microbit.org/).

8 Métropole, juin 2021, série professionnelle

Références

Série : générale

Académie : Métropole-La Réunion

Date : juin 2021

Exercice 5 - 14 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs

Thèmes informatiques

- itération bornee
- bloc sans paramètre

Compétences travaillées

- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	1
calcul	0
aléatoire	1
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

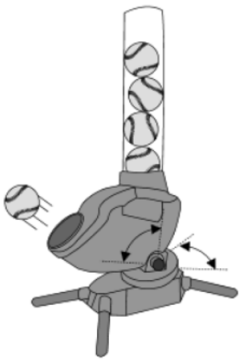
Énoncé - Métropole professionnel, 28 juin 2021

Pour s'entrainer, les batteurs de base-ball utilisent une machine lance-balles. Cette machine définit aléatoirement trois paramètres :

- la vitesse d'envoi de la balle en km/h ;
- l'angle en degré dans le plan horizontal ;
- l'angle en degré dans le plan vertical

Cette machine utilise un programme dont le script est ci-dessous.

```
quand [drapeau] est cliqué
répéter 10 fois
    mettre vitesse à nombre aléatoire entre 32 et 170
    mettre anglehorizontal à nombre aléatoire entre -10 et 10
    mettre anglevertical à nombre aléatoire entre 0 et 20
    Lancer la balle
    attendre 5 secondes
```



1. Rédiger une phrase précisant l'intervalle de vitesse avec laquelle est lancée la balle.
   On souhaite modifier le script pour effectuer seulement une série de 5 lancers mais toutes les 4 secondes avec un angle horizotal allant de -5 à 20 degrés.
2. Compléter le script en **ANNEXE 2 à rendre avec la copie**.

ANNEXE 2

```
quand [drapeau] est cliqué
répéter ( ) fois
    mettre vitesse à nombre aléatoire entre 32 et 170
    mettre anglehorizontal à nombre aléatoire entre ( ) et ( )
    mettre anglevertical à nombre aléatoire entre 0 et 20
    Lancer la balle
    attendre ( ) secondes
```

Correction

Fichier Scratch : `pro_metropole_2021_juin.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/578344184/`

1. L'intervalle de vitesse avec laquelle est lancée la balle va de 32 à 170 unités.
2. **ANNEXE 2**

Analyse

Commentaires de l'IREM de Marseille : Cet exercice présente peu d'intérêt informatique.
L'exercice revient à "extraire l'information utilsée" sans rien d'autre....

Brevets 2022

1 Polynésie, juin 2022, série générale, exercice 5

Références

Série : générale

Académie : Polynésie

Date : juin 2022

Exercice 5 - 23 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- frise

Thèmes informatiques

- activité du lutin
- itération bornée

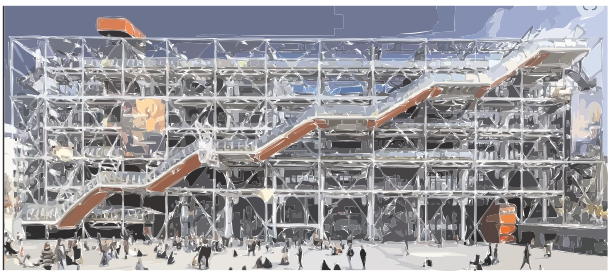
Compétences travaillées

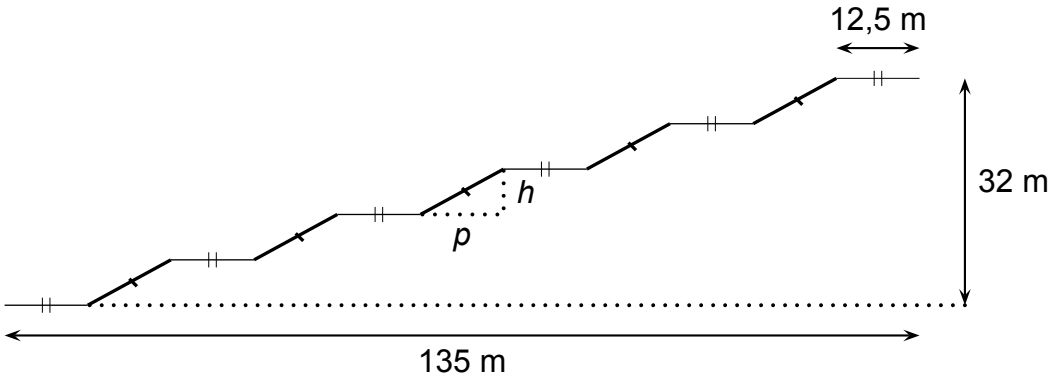
- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, 23 juin 2022, Exercice 5

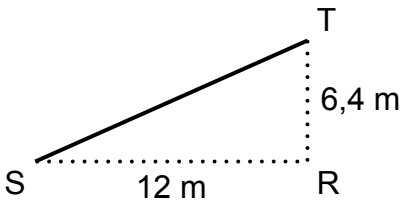
Le centre Pompidou est un musée d'art contemporain à Paris. Pour accéder aux étages, il faut utiliser un ensemble d'escalators extérieurs appelé « chenille ».
La chenille est composée de 5 escalators tous identiques (traits épais sur la figure ci-dessous) et de 6 passerelles horizontales toutes identiques (traits fins horizontaux sur la figure ci-dessous).



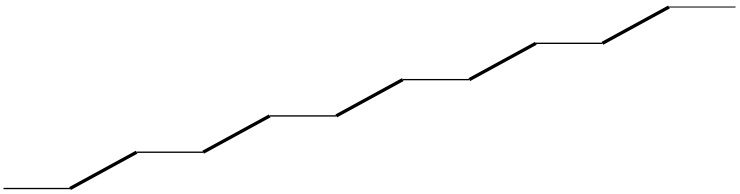


1. À l'aide de la figure ci-dessus :
   (a) Vérifier que la profondeur p de chaque escalator est égale à 12 m
   (b) Calculer la hauteur h de chaque escalator

2. À l'aide du triangle RST ci-contre :
   (a) Prouver que la longueur ST d'un escalator est de 13,6 m.
   (b) Montrer que la mesure de l'angle formé par l'escalator avec l'horizontale (c'est-à-dire l'angle $\widehat{RST}$ arrondie au degré est de 28 ° .



3. Sabine veut représenter la chenille grâce au logiciel Scratch.
   Elle a écrit le programme qui est donné sur l'ANNEXE. On précise que : 1 pas du logiciel correspond à 1 m dans la réalité.
   Compléter les lignes 6, 7, 9, et 10, sur l'ANNEXE (à rendre avec la copie), afin d'obtenir le tracé ci-dessous de la chenille :

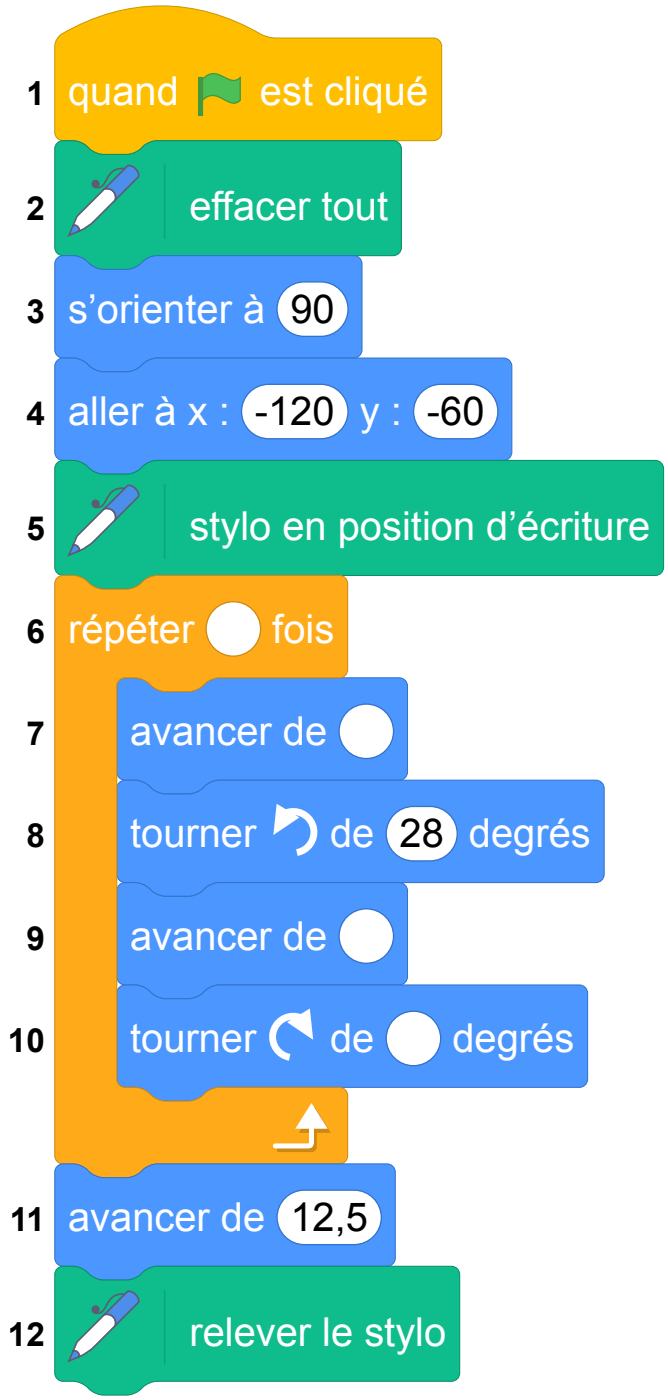


Rappel : « S'orienter à 90 » signifie que l'on est orienté vers la droite

ANNEXE

À compléter et à rendre avec la copie

Exercice 5 question 3 :



Correction

Fichier scratch : `sg_polynesie_2021_juin.sb3` Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/711337246/`

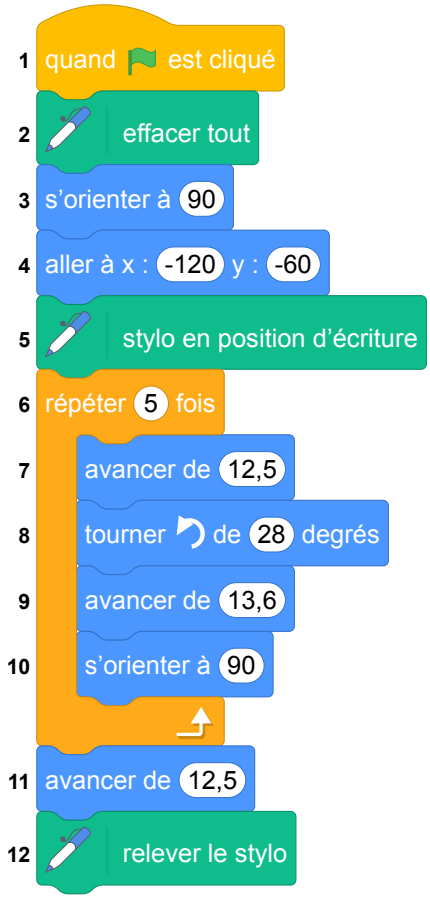
1. (a) $135 = 6 \times 12,5 + 5p$, soit $135 = 75 + 5p$, d'où en ajoutant -75 à chaque membre :
   $135 - 75 = 5p$ ou $60 = 5p$, c'est-à-dire $5 \times 12 = 5 \times p$, d'où $p = 12$ m.

   (b) On a 5 hauteurs de h pour un total de 32 m, soit : $5 \times h = 32$, d'où $h = \dfrac{32}{5} = \dfrac{64}{10} = 6,4$ (m).
   La hauteur de chaque escalator est de 6,4 m.

2. (a) Dans le triangle RST, rectangle en R on utilise le théorème de Pythagore, soit :
   $ST^2 = SR^2 + RT^2 = 12^2 + 6,4^2 + = 144 + 40,96 = 184,96$.
   D'où $ST = \sqrt{184,96} = 13,6$ m.

   (b) Dans le triangle RST rectangle en R on a d'après la trigonométrie :
   $\cos \widehat{RST} = \dfrac{SR}{ST} = \dfrac{12}{13,6} \approx 0,882$.
   La calculatrice donne $\widehat{RST} \approx 28,07$ soit 28° au degré près.

3. Script complété :



2 Asie, juin 2022, série générale

Références

Série : générale

Académie : Asie

Date : juin 2022

Exercice 4 - 25 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- calcul sans programme

Thèmes informatiques

- variable réservée
- calcul
- branchement conditionnel

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	1
complète (si-alors-sinon)	1
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Asie, juin 2022

Une boutique en ligne vend des photos et affiche les tarifs suivants :

Nombre de photos commandées	Prix à payer
De 1 à 100 photos	0,17 € par photo
Plus de 100 photos	17 € pour l'ensemble des 100 premières photos et 0,13 € par photo supplémentaire

1. (a) Quel est le prix à payer pour 35 photos ?
   (b) Vérifier que le prix à payer pour 150 photos est 23,50 €.
   (c) On dispose d'un budget de 10 €. Combien de photos peut-on commander au maximum ?

On a commencé à construire un programme qui doit permettre de calculer le prix à payer en fonction du nombre de photos commandées :

```
1 quand [drapeau] est cliqué
2 demander [Nombre de photos à commander ?] et attendre
3 mettre [Nb photos] à [réponse]
4 si <[Nb photos] < ( )>
5     mettre [Prix] à [Nb photos] * ( )
6 sinon
7     mettre [Nb photos supplémentaires] à [Nb photos] - [100]
8     mettre [Prix] à ( ) + [Nb photos supplémentaires] * [0.13]
9 dire [regrouper [Prix à payer en euros] et [Prix]]
```

Informations :
Le programme comporte trois variables :

- Nb photos
  Nombre de photos commandées
- Nb photos supplémentaires
  Nombre de photos commandées au-delà des 100 premières photos commandées.
- Prix

2. *Dans cette question, aucune justification n'est attendue.*
   Par quelles valeurs peut-on compléter les instructions des lignes 4, 5 et 8 pour que le programme permette de calculer le prix à payer en fonction du nombre de photos commandées ?
   Sur la copie, écrire le numéro de chaque ligne à compléter et la valeur correspondante.
3. En période des soldes, le site offre une réduction de 30 % sur le prix à payer, pour toute commande supérieure à 20 €.
   (a) Calculer le prix a payer pour 150 photos en période des soldes.
   (b) *Dans cette question, aucune justification n'est attendue.*

   On modifie le programme pour qu'il donne le prix à payer en période des soldes en insérant le bloc ci-contre entre les lignes 8 et 9.
   Dans la liste suivante, indiquer une proposition qui convient pour compléter la case vide :

```
si <[Prix] > [20]> alors
    mettre [Prix] à ( )
```

Proposition 1 : [Prix] - [30]
Proposition 2 : [Prix] - [Prix] * [0.3]
Proposition 3 : [Prix] * [30] / [100]
Proposition 4 : [Prix] * [0.7]

Correction

Fichier Scratch : `sg_asie_2022_juin.sb3` Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/711334914/`

1. (a) On calcule $35 \times 0,17 = 5,95$€ : 35 photos coûtent 5,95 €.
   (b) $17 + 50 \times 0,13 = 17 + 6,5 = 23,50$ €.
   (c) Nous devons effectuer la division de 10 par 0,17 :
   $10 = 0,17 \times 58 + 0,14$.
   Avec un budget de 10 € on ne pourra commander que 58 photos.
2. ligne 4 : 101 Ligne 5 : 0,17 Ligne 8 : 17
3. (a) $23,50 \times (1 - 0,3) = 23,50 \times 0,7 = 16,45$ (€).
   En période, de soldes le prix de 150 photos sera de 16,45 €.
   (b) Retrancher 30 %, c'est multiplier par $\left(1 - \frac{30}{100}\right) = 1 - 0,3 = 0,7$: les propositions 2 et 4 conviennent.

3 Métropole, La Réunion, Antilles-Guyanne (Secours), juin 2022, série générale

Références

Série : générale

Académie : Métropole, La Réunion, Antilles, Guyanne (Secours)

Date : juin 2022

Exercice 4 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- calcul sans programme
- fonctions
- inéquations

Thèmes informatiques

- variable réeservee
- variable déclaree
- calcul

Compétences travaillées

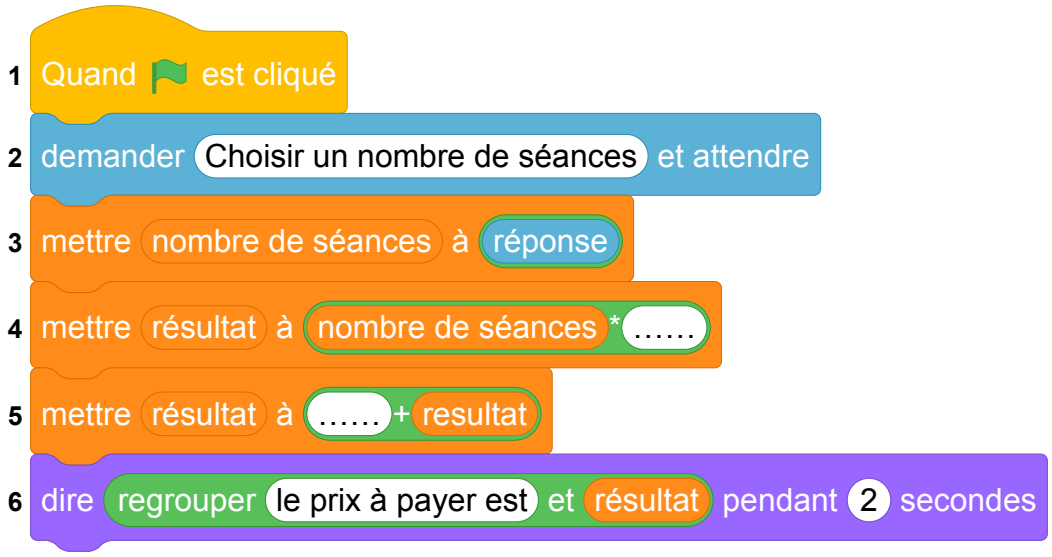
- compléter un algorithme
- utiliser une variable avec auto référence

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion – Antilles-Guyane (Secours), 30 juin 2022

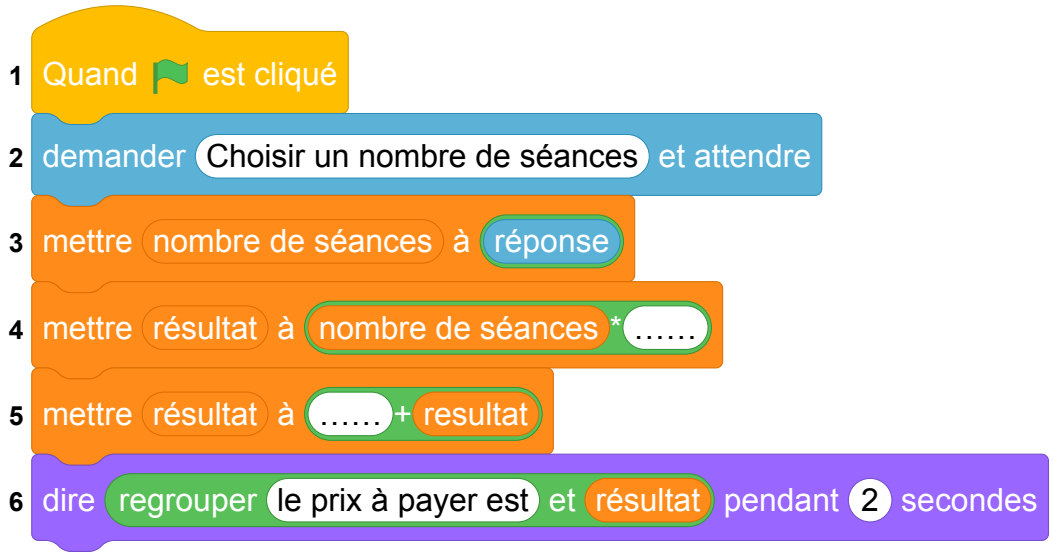
Un club de sport propose une nouvelle formule annuelle pour ses adhérents :
« Achat d'une carte d'adhésion à 32 € donnant droit à un tarif de 4,50 € par séance ».

1. Déterminer le coût à payer pour dix séances dans l'année avec cette formule.
2. Noé a un budget annuel de 95 € pour se rendre dans cette salle de sport.
   Combien de séances pourrait-il effectuer ?
3. On note p la fonction qui, au nombre x de séances pratiquées, associe le prix à payer pour x séances pratiquées dans l'année.
   (a) Donner l'expression de $p(x)$.
   (b) Vérifier que $p(27) = 153,5$.
   (c) Interpréter par une phrase l'égalité précédente.
4. On s'intéresse au programme qui permet de donner le prix à payer en fonction du nombre de séances pratiquées dans cette salle de sport.



Compléter sur l'ANNEXE les lignes 4 et 5 pour que ce script corresponde au programme souhaité.

ANNEXE



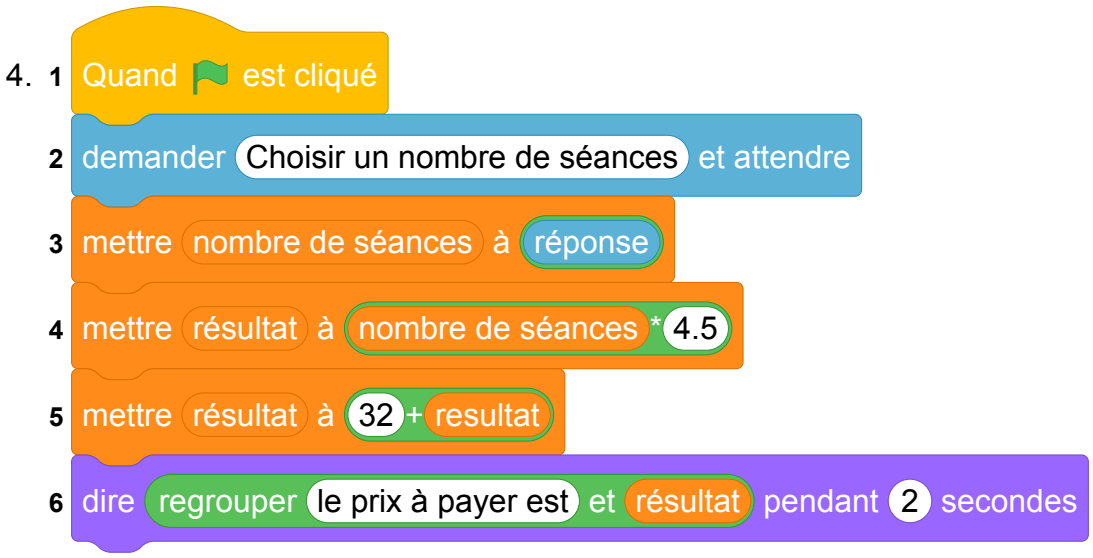
Correction

Fichier scratch : `sg_metropole_2022_juin_secours.sb3`

Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/791811847/`

1. Le coût à payer pour dix séances dans l'année avec cette formule est $32+10\times4,5 = 32+45 = 77$ €
2. Si x est le nombre de séances, le coût total est $32 + x \times 4,5$.
   Ce coût doit être inférieur ou égal à 95 € c'est à dire que
   $32 + x \times 4,5 \leq 95$
   $32 + x \times 4,5 - 32 \leq 95 - 32$
   $32 + x \times 4,5 - 32 \leq 95 - 32$
   $x \times 4,5 \leq 63$
   $x \times 4,5\ div 4,5 \leq 63\ div 4,5$
   $x \leq 14$
   Le nombre maximal de séances auquelles il peut assister est donc 14.
3. On note p la fonction qui, au nombre x de séances pratiquées, associe le prix à payer pour x séances pratiquées dans l'année.
   (a) $p(x) = 32 + x \times 4,5$.
   (b) $p(27) = 32 + 27 \times 4,5 = 32 + 121,5 = 153,5$
   (c) Le coût de 27 séances avec ce tarif est de $153,5$ €

   ANNEXE

4.


4 Métropole, La Réunion, Antilles-Guyanne, juin 2022, série générale

Références

Série : générale

Académie : Métropole, La Réunion, Antilles, Guyanne

Date : juin 2022

Exercice 4 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable réservée
- variable déclarée
- calcul

Compétences travaillées

- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

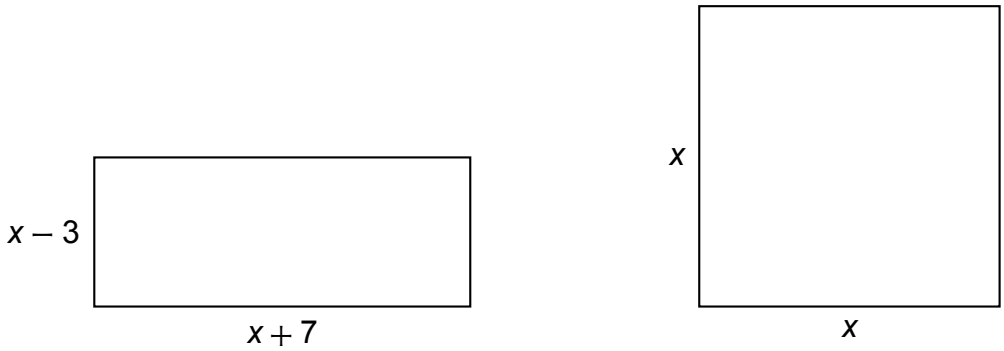
Énoncé - Métropole – La Réunion – Antilles-Guyane, 30 juin 2022

Dans cet exercice, x est un nombre strictement supérieur à 3.

On s'intéresse aux deux figures géométriques dessinées ci-dessous :

— un rectangle dont les côtés ont pour longueurs $x - 3$ et $x + 7$.

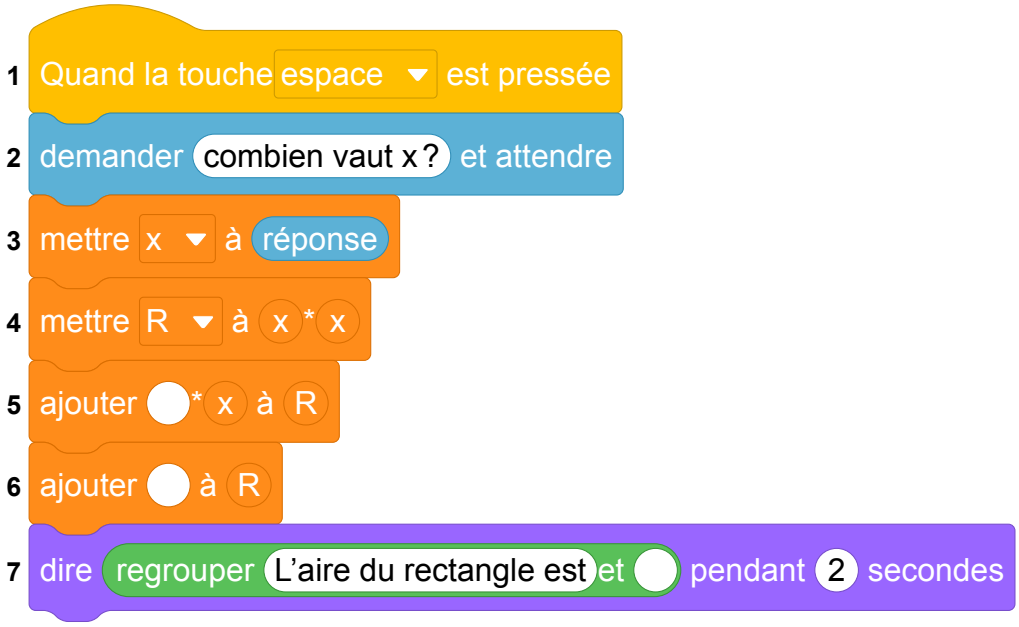
— un carré de côté x.



1. Quatre propositions sont écrites ci-dessous :
   Recopier sur la copie celle qui correspond à l'aire du carré. On ne demande pas de justifier.

$4x$	$4 + x$	x^2	$2x$

2. Montrer que l'aire du rectangle est égale à $x^2 + 4x - 21$.
3. On a écrit le script ci-dessous dans Scratch.
   On veut que ce programme renvoie l'aire du rectangle lorsque l'utilisateur a rentré une valeur de x (strictement supérieure à 3).
   Écrire sur la copie les contenus des trois cases vides des lignes 5, 6 et 7, en précisant les numéros de lignes qui correspondent à vos réponses.

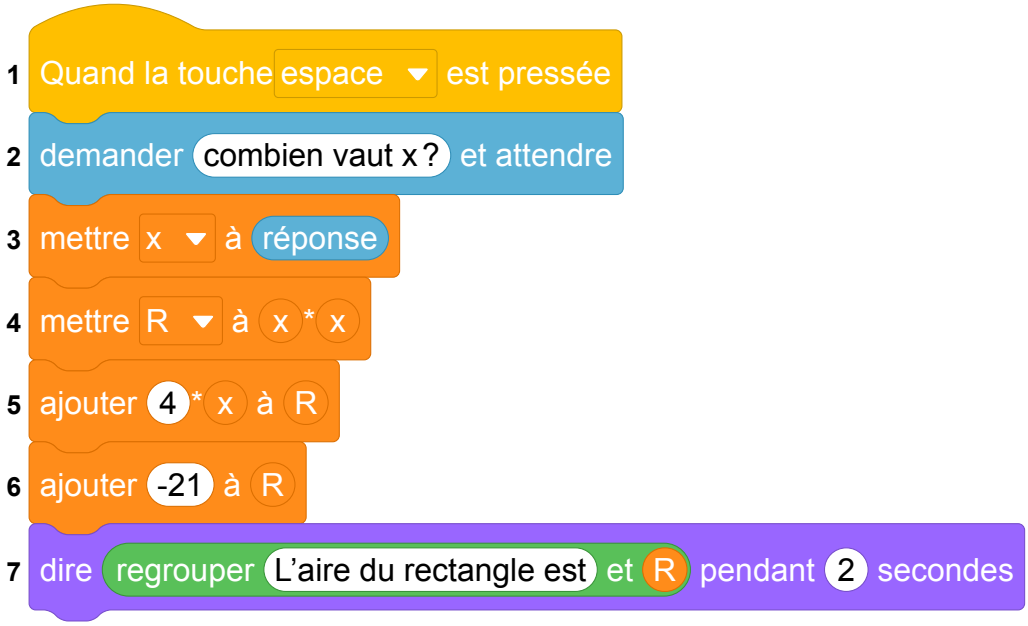


4. On a pressé la touche espace puis saisi le nombre 8. Que renvoie le programme ?
5. Quel nombre x doit-on choisir pour que l'aire du rectangle soit égale à l'aire du carré ?
   Toute trace de recherche, même non aboutie, sera prise en compte.

Correction

Fichier scratch : `sg_metropole_2022_juin.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/711269248/`

1. L'aire du carré est x^2.
2. L'aire du rectangle est égale au produit de sa longueur par sa largeur soit à $(x-3)(x+7) = x^2 + 7x - 3x - 21 = x^2 + 4x - 21$.
3. Le script en Scratch complété est



4. On a pressé la touche espace puis saisi le nombre 8.
   — Ligne 3 : la variable x prend la valeur 8
   — Ligne 4 : la variable R prend la valeur $x^2 = 8^2 = 64$
   — Ligne 5 : On ajoute $4 \times x = 4 \times 8 = 32$ à la variable R qui vaut 64.
   R prend donc la valeur $= 64 + 32 = 96$
   — Ligne 6 : On ajoute -21 à la variable R qui vaut 96.
   R prend donc la valeur $= 96 - 21 = 75$
5. Pour trouver le nombre x pour que l'aire du rectangle soit égale à l'aire du carré on résoud l'équation $x^2 + 4x - 21 = x^2$.
   On soustrait x^2 des deux côtés de l'égalité ce qui donne $4x - 21 = 0$.
   On ajoute 21 des deux côtés $4x - 21 + 21 = 0 + 21$ soit $4x = 21$.
   On divise par 4 des deux côtés $x = \dfrac{21}{4} = 5,25$

5 Amérique du Nord, juin 2022, série générale

Références

Série : générale

Académie : Amérique du Nord

Date : juin 2022

Exercice 4 - 21 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- grandeurs et mesures
- probabilités
- proportionnalité

Thèmes informatiques

- état

Compétences travaillées

- exécuter un algorithme
- écrire un algorithme

Compétences travaillées

- exécuter un algorithme
- écrire un algorithme

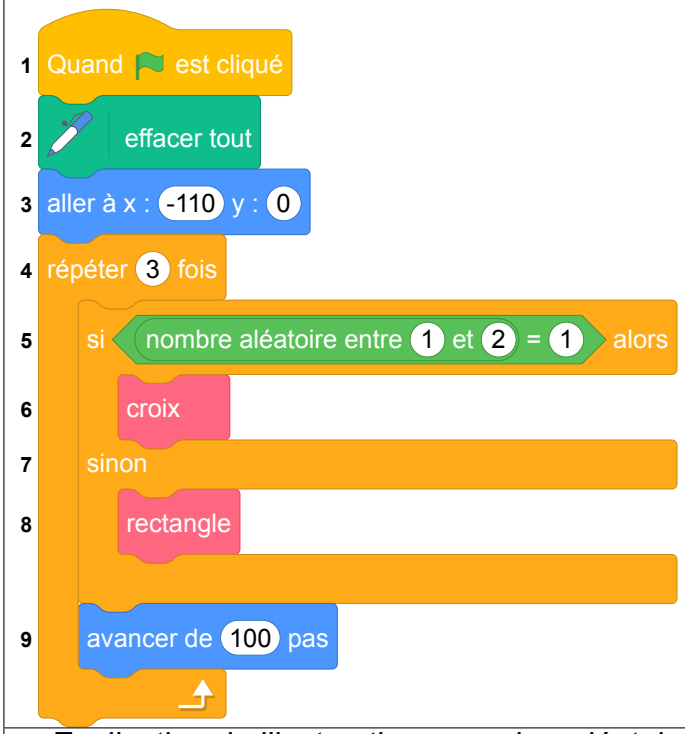
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	1
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	1
itération bornée	1
itération conditionnelle	0
imbrication de structures	1
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	1
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Amérique du Nord, 3 juin 2022

Dans cet exercice, aucune justification n'est attendue.

On a créé un jeu de hasard à l'aide d'un logiciel de programmation.
Lorsqu'on appuie sur le drapeau, le lutin dessine trois motifs côte à côte.
Chaque motif est dessiné aléatoirement : soit c'est une croix, soit c'est un rectangle.
Le joueur gagne si l'affichage obtenu comporte trois motifs identiques.
Au lancement du programme, le lutin est orienté horizontalement vers la droite :

Programme principal

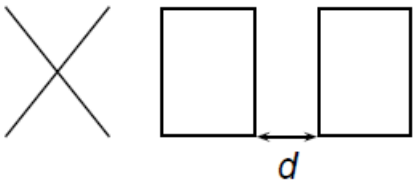


Bloc « rectangle »

Bloc « croix »
Le script n'est pas donné.

Explication de l'instruction « nombre aléatoire entre ... » sur un exemple :
nombre aléatoire entre 1 et 4 renvoie un nombre au hasard parmi 1, 2, 3 et 4.

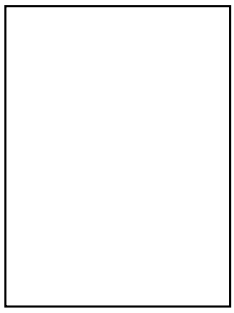
1. En prenant pour échelle 1 cm pour 20 pas, représenter le motif obtenu par le bloc « rectangle ».
2. Voici un exemple d'affichage obtenu en exécutant le programme principal :
   Quelle est la distance d entre les deux rectangles sur l'affichage, exprimée en pas ?



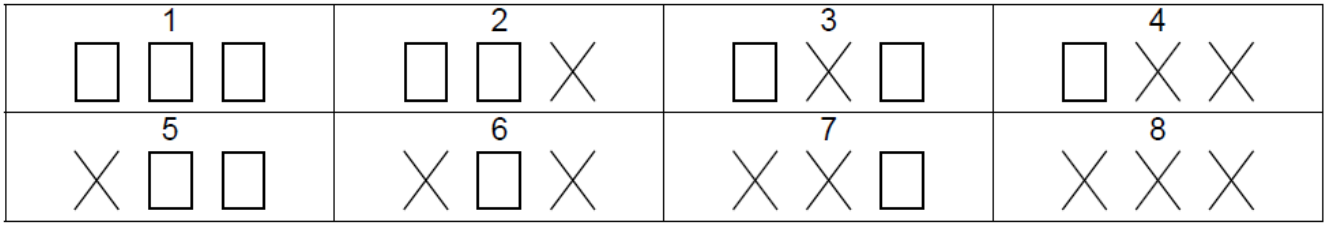
3. Quelle est la probabilité que le premier motif dessiné par le lutin soit une croix ?
4. Dessiner à main levée les 8 affichages différents que l'on pourrait obtenir avec le programme principal.
5. On admettra que les 8 affichages ont la même probabilité d'apparaître. Quelle est la probabilité que le joueur gagne ?
6. On souhaite désormais que, pour chaque motif, il y ait deux fois plus de chances d'obtenir un rectangle qu'une croix. Pour cela, il faut modifier l'instruction dans la ligne 5.
   Sur la copie, recopier l'instruction suivante en complétant les cases :

Correction

Fichier scratch : `sg_amerique_du_nord_2022_juin.sb3` Lien vers le fichier Scratch en ligne : `scratch.mit.edu/projects/711331683/`



1. Le rectangle fait 60 pas horizontalement (le lutin est orienté horizontalement vers la droite au début), donc 3 cm de large et 80 pas verticalement, donc 4 cm de haut. On doit donc représenter le rectangle ci-contre.
2. En analysant le bloc rectangle, on a compris qu'il faisait 60 pas de large. À la fin de l'exécution, le lutin est revenu à son point de départ (le coin en bas à gauche du rectangle), avec son orientation de départ (orienté horizontalement vers la droite), et dans le programme principal (ligne 9), on voit que le lutin avance de 100 pas avant de recommencer à tracer, soit un rectangle, soit une croix.
   La distance entre deux motifs est donc $d = 100 - 60 = 40$ pas.
3. Le premier motif dessiné par le lutin est une croix si le nombre aléatoire entre 1 et 2 est 2.
   On a donc une probabilité de $\frac{1}{2} = 0,5$ que cela arrive.
4. On obtient les huit possibilités suivantes :



5. Si les 8 affichages ont la même probabilité d'apparaître, sachant que deux affichages correspondent à la victoire (les affichages 1 et 8), la probabilité que le joueur gagne est donc de $\frac{2}{8} = \frac{1}{4} = 0,25$.
6. Pour qu'il y ait deux fois plus de chances d'obtenir un rectangle qu'une croix, il faut que les probabilités d'apparaître soient $\frac{2}{3}$ pour le rectangle et $\frac{1}{3}$ pour la croix.
   Pour cela, il faut modifier l'instruction dans la ligne 5 en :

6 Polynésie, juin 2022, série générale, exercice 4

Références

Série : générale

Académie : Polynésie

Date : juin 2022

Exercice 4 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- nombres et calculs
- programme de calcul

Thèmes informatiques

- variable déclarée
- activité du lutin
- calcul
- itération bornée
- bloc sans paramètre

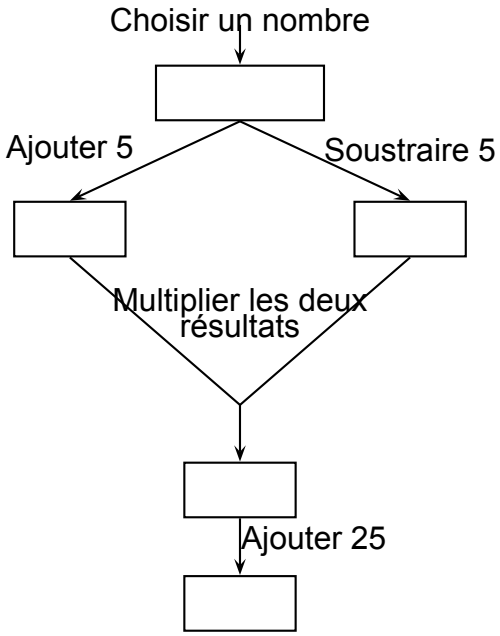
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, 23 juin 2022, exercice 4

On considère le programme de calcul suivant :



1. (a) Si on choisit le nombre 7, vérifier qu'on obtient 49 à la fin du programme
   (b) Si on choisit le nombre -4, quel résultat obtient-on à la fin du programme ?
2. On note x le nombre choisi au départ
   (a) Exprimer en fonction de x le résultat obtenu.
   (b) Développer et réduire $(x+5)(x-5)$.
   (c) Sarah dit : « Avec ce programme de calcul, quel que soit le nombre choisi au départ, le résultat obtenu est toujours le carré du nombre de départ ».
   Qu'en pensez-vous ?

Correction

1. (a) $(7+5)\times(7-5)+25 = 12\times 2+25 = 24+25 = 49$.
   Avec 5 au départ on obtient bien 49 en sortie.

   (b) $(-4+5)(-4-5)+25 = 1\times(-9)+25 = -9+25 = 16$.
   Avec -4 au départ on obtient 16 en sortie.

2. (a) $(x+5)(x-5)+25$

   (b) On développe $(x+5)(x-5) = x^2-5^2 = x^2-25$.

   (c) D'après le calcul précédent : $(x+5)(x-5)+25 = x^2-25+25 = x^2$. Sarah a raison.

7 Centres étrangers, juin 2022, série générale

Références

Série : générale générale

Académie : Centres étrangers

Date : juin 2022

Exercice 4 - 15 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- frise
- rosace
- déplacement
- translation
- rotation

Thèmes informatiques

- activité du lutin
- calcul
- itération bornée
- bloc sans parametre

Compétences travaillées

- Aucune compétence algorithmique

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Centres étrangers, 14 Juin 2022

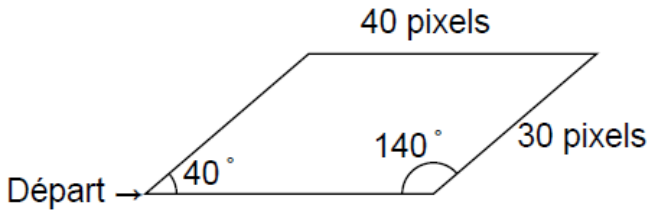
Dans cet exercice, toutes les longueurs sont exprimées en pixel.

Partie A :

Un professeur donne à ses élèves un motif en forme de parallélogramme et le script, en partie rédigé, qui permet de tracer ce motif.

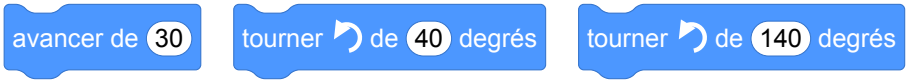
On précise que le lutin est au point de départ, comme indiqué sur la figure ci- dessous, et qu'il est orienté vers la droite :

Parallélogramme obtenu :



Script du motif

Recopier dans le bon ordre, sur votre copie, les instructions suivantes à insérer dans le script du motif permettant de tracer le parallélogramme ci-dessus :

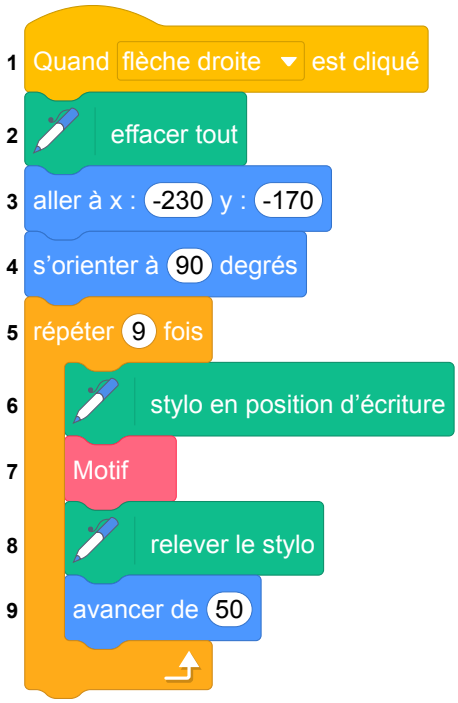


Partie B :

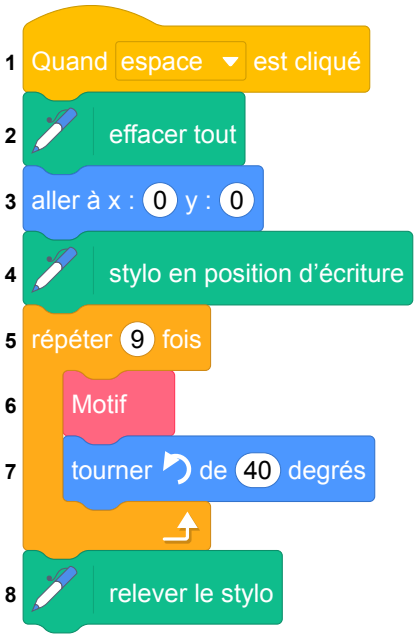
Le professeur demande ensuite à ses élèves d'intégrer ce script dans un programme de leur choix permettant de tracer des figures composées de plusieurs de ces motifs.

Voici les programmes écrits par deux élèves.

Programme de l'élève A

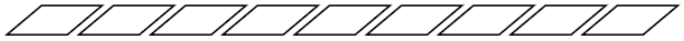
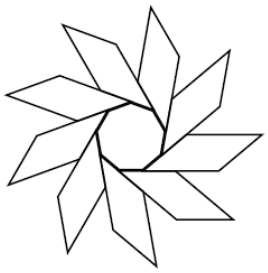
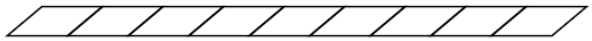
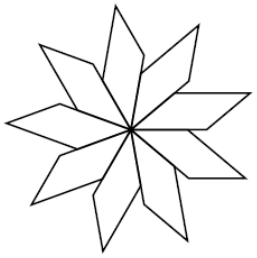


Programme de l'élève B



On rappelle que « s'orienter à 90 » signifie que l'on est orienté vers la droite.

1. Quelle action au clavier permet de lancer le programme de l'élève B ?
2. Parmi les figures suivantes, indiquer, ici **sans justifier** :
   (a) laquelle est obtenue avec le programme de l'élève A ?
   (b) laquelle est obtenue avec le programme de l'élève B ?

Figure 1 	**Figure 2** 
Figure 3 	**Figure 4** 

Correction

Fichier scratch : `sg_centres_etrangers_2022_juin.sb3`

Lien vers le fichier Scratch en ligne : `scratch.mit.edu/projects/711333585/`

Partie A

Un professeur donne à ses élèves un motif en forme de parallélogramme et le script, en partie rédigé, qui permet de tracer ce motif.

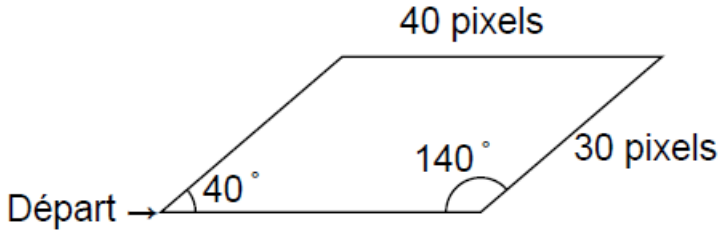
On précise que le lutin est au point de départ, comme indiqué sur la figure ci- dessous, et qu'il est orienté vers la droite :

Parallélogramme obtenu :

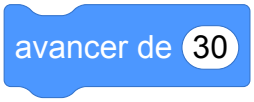
Script du motif

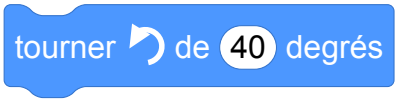
Parallélogramme obtenu :

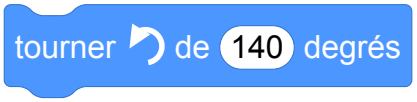
Script du motif



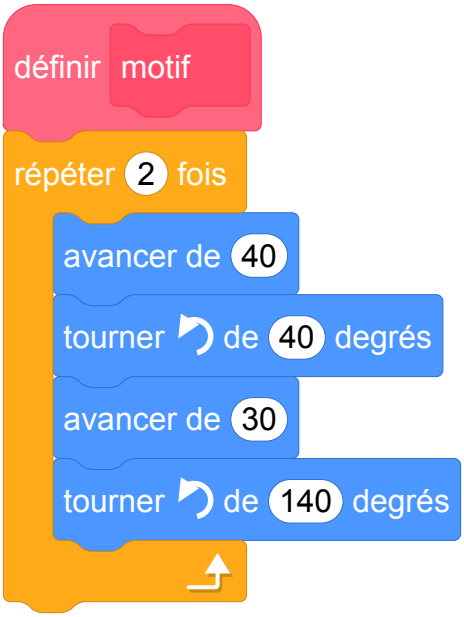
On considère les instructions :







Avec ces instructions, on complète le script du motif permettant de tracer le parallélogramme :



Partie B

Le professeur demande ensuite à ses élèves d'intégrer ce script dans un programme de leur choix permettant de tracer des figures composées de plusieurs de ces motifs.

Voici les programmes écrits par deux élèves.

Programme de l'élève A

```
1 Quand flèche droite est cliqué
2 effacer tout
3 aller à x : -230 y : -170
4 s'orienter à 90 degrés
5 répéter 9 fois
6     stylo en position d'écriture
7     Motif
8     relever le stylo
9     avancer de 50
```

Programme de l'élève B

```
1 Quand espace est cliqué
2 effacer tout
3 aller à x : 0 y : 0
4 stylo en position d'écriture
5 répéter 9 fois
6     Motif
7     tourner de 40 degrés
8 relever le stylo
```

On rappelle que « s'orienter à 90 » signifie que l'on est orienté vers la droite.

1. Pour lancer le programme de l'élève B il faut appuyer sur la barre d'espace.
2. Parmi les figures suivantes :
   (a) la figure 1 est obtenue avec le programme de l'élève A ;
   (b) la figure 4 est obtenue avec le programme de l'élève B.

Figure 1	Figure 2

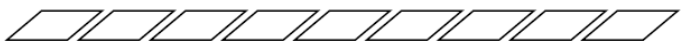
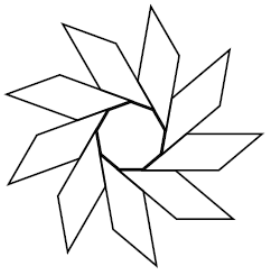
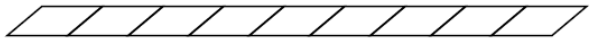
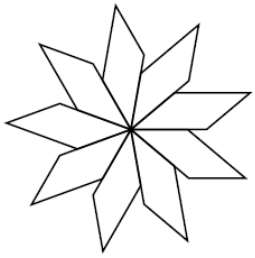

	

Figure 3	**Figure 4**
	

8 Polynésie, septembre 2022, série générale

Références

Série : générale

Académie : Polynésie

Date : septembre 2022

Exercice 3 - 17 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- repérage
- déplacement
- symétrie
- translation

Thèmes informatiques

- variable déclarée
- activité du lutin
- itération bornée

Compétences travaillées

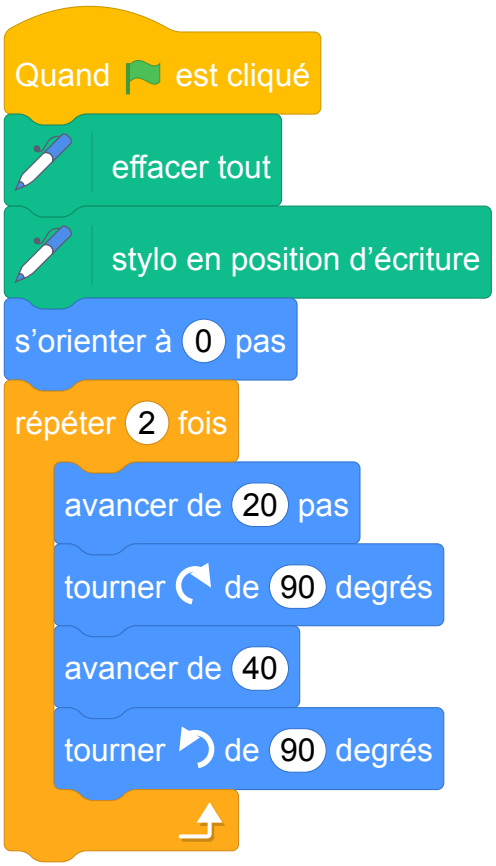
- comprendre un algorithme
- exécuter un algorithme
- modifier un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	0
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	1
chgt avec auto-référence	1
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Polynésie, 6 septembre 2022

On utilise un logiciel de programmation.
On rappelle que « s'orienter à 0 ° » signifie qu'on oriente le stylo vers le haut.
On considère les deux scripts suivants :

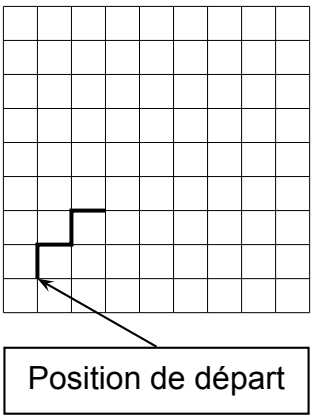
Script 1



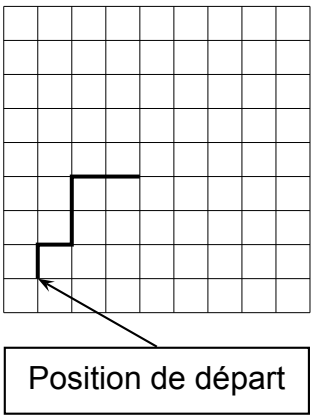
Script 2

1. On exécute le script 1 ci-dessus.
   Représenter le chemin parcouru par le stylo sur l'ANNEXE à rendre avec la copie.
2. Quel dessin parmi les trois ci-dessous correspond au script 2 ?
   On expliquera pourquoi les deux autres dessins ne correspondent pas au script 2.
   Chaque côté de carreau mesure 20 pixels.

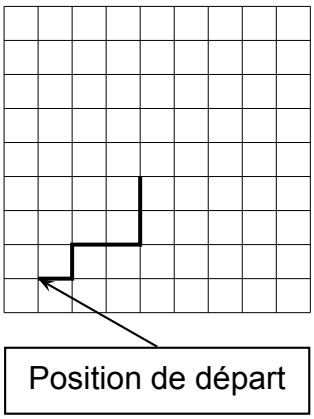
Dessin 1



Dessin 2

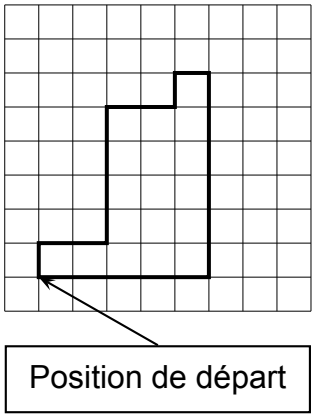


Dessin 3



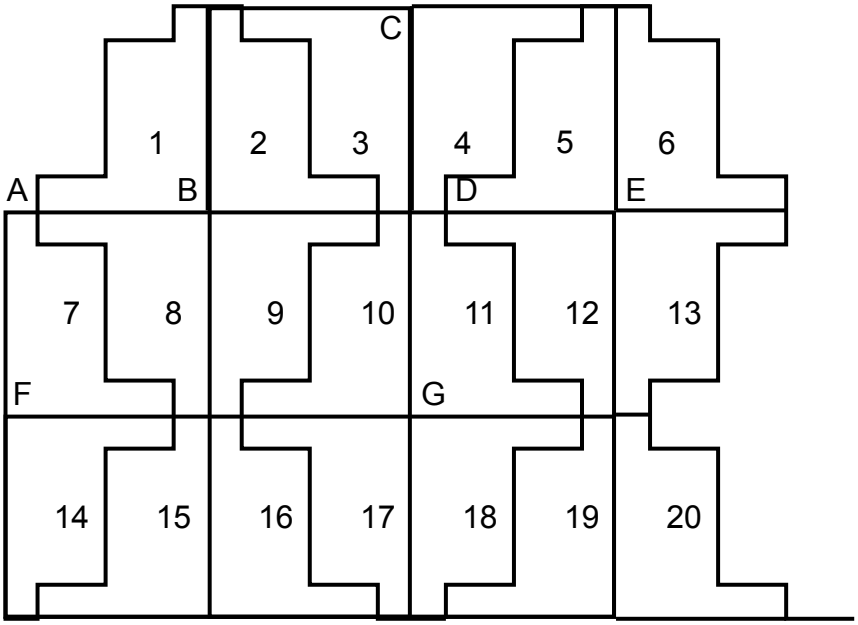
3. On souhaite maintenant obtenir le motif représenté sur le dessin 4 :

Dessin 4



Compléter sans justifier les trois cases du script 3 donné en ANNEXE à rendre avec la copie, permettant d'obtenir le dessin 4.

4. À partir du motif représenté sur le dessin 4, on peut obtenir le pavage ci-dessous :



Répondre aux questions suivantes sur votre copie en indiquant le numéro du motif qui convient (on ne demande pas de justifier la réponse) :

(a) Quelle est l'image du motif 1 par la translation qui transforme le point B en E ?

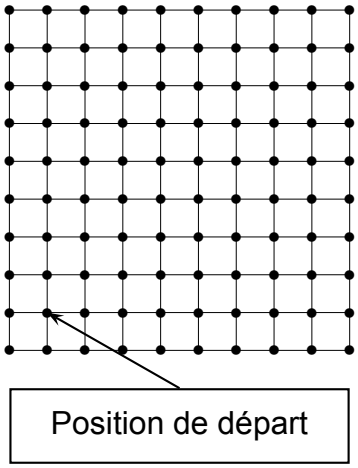
(b) Quelle est l'image du motif 1 par la symétrie de centre B ?

(c) Quelle est l'image du motif 16 par la symétrie de centre G ?

(d) Quelle est l'image du motif 2 par la symétrie d'axe (CG) ?

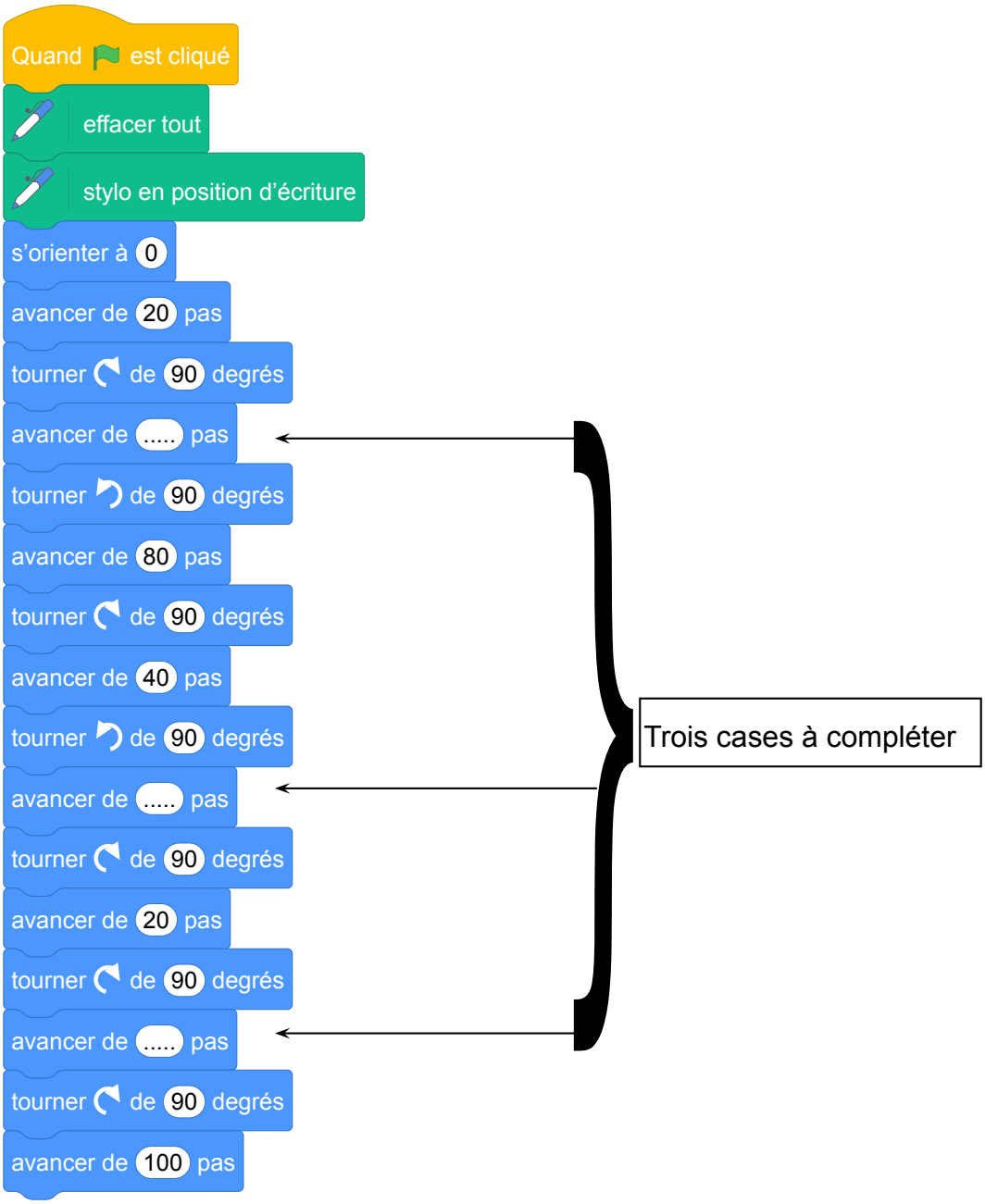
ANNEXE

Exercice 3. Question 1



Chaque côté de carreau mesure 20 pixels.
La position de départ du stylo est indiquée sur la figure ci-contre.

Exercice 3. Question 3



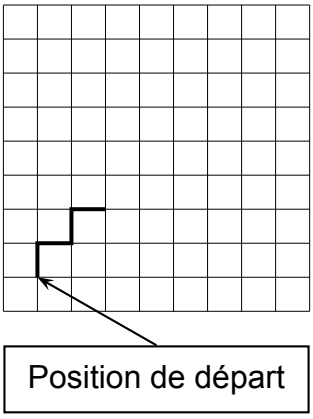
Correction

Fichier scratch : `sg_polynesie_2022_septembre.sb3`

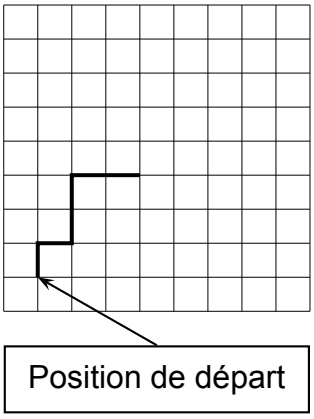
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/775368495/`

1. On exécute le script 1 ci-dessus.
   Représenter le chemin parcouru par le stylo sur l'ANNEXE à rendre avec la copie.
   Le tracé est en rouge sur l'ANNEXE.
2. Le dessin 1 n'est pas correct car après avoir avancé deux fois de 20 on doit avancer de 40.
   Le dessin 3 n'est pas correct car on ne se dirige pas au départ vers le haut.
   Il reste donc le dessin 2 seul correct.

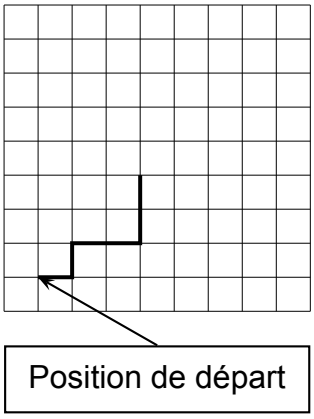
Dessin 1



Dessin 2

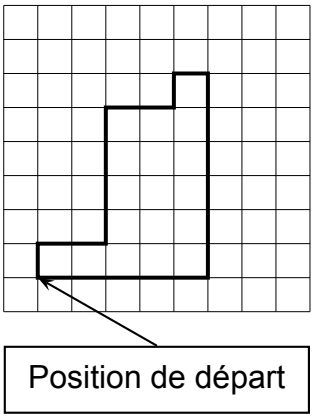


Dessin 3



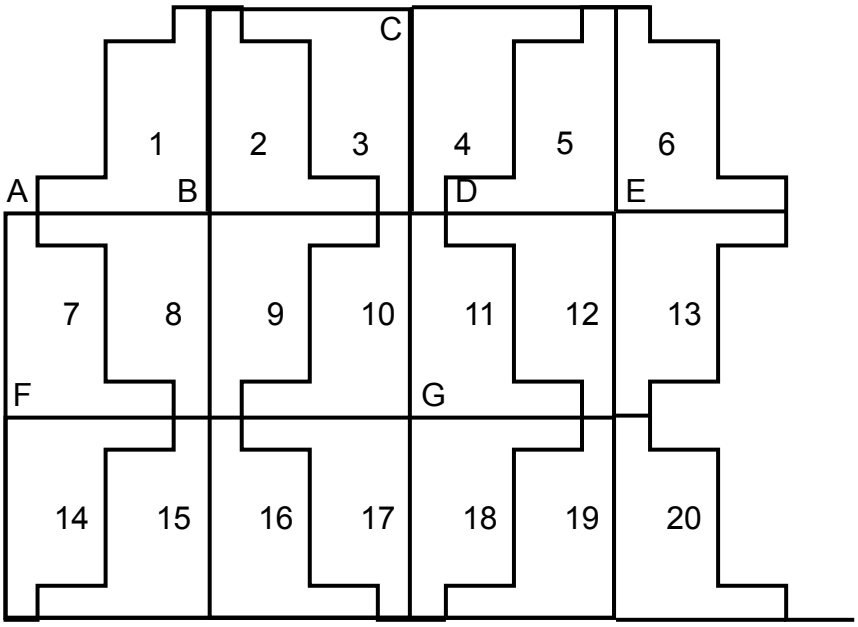
3. On souhaite maintenant obtenir le motif représenté sur le dessin 4 :

Dessin 4



   Compléter sans justifier les trois cases du script 3 donné en ANNEXE à rendre avec la copie, permettant d'obtenir le dessin 4.
   Les compléments sont en rouge dans l'annexe.

4. À partir du motif représenté sur le dessin 4, on peut obtenir le pavage ci-dessous :



Répondre aux questions suivantes sur votre copie en indiquant le numéro du motif qui convient (on ne demande pas de justifier la réponse) :

(a) Quelle est l'image du motif 1 par la translation qui transforme le point B en E ? Le motif 5.

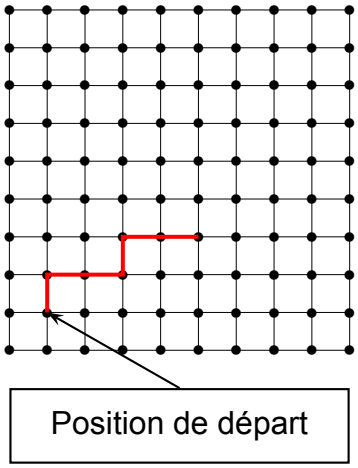
(b) Quelle est l'image du motif 1 par la symétrie de centre B ? Le motif 9.

(c) Quelle est l'image du motif 16 par la symétrie de centre G ? Le motif 12.

(d) Quelle est l'image du motif 2 par la symétrie d'axe (CG) ? Le motif 5.

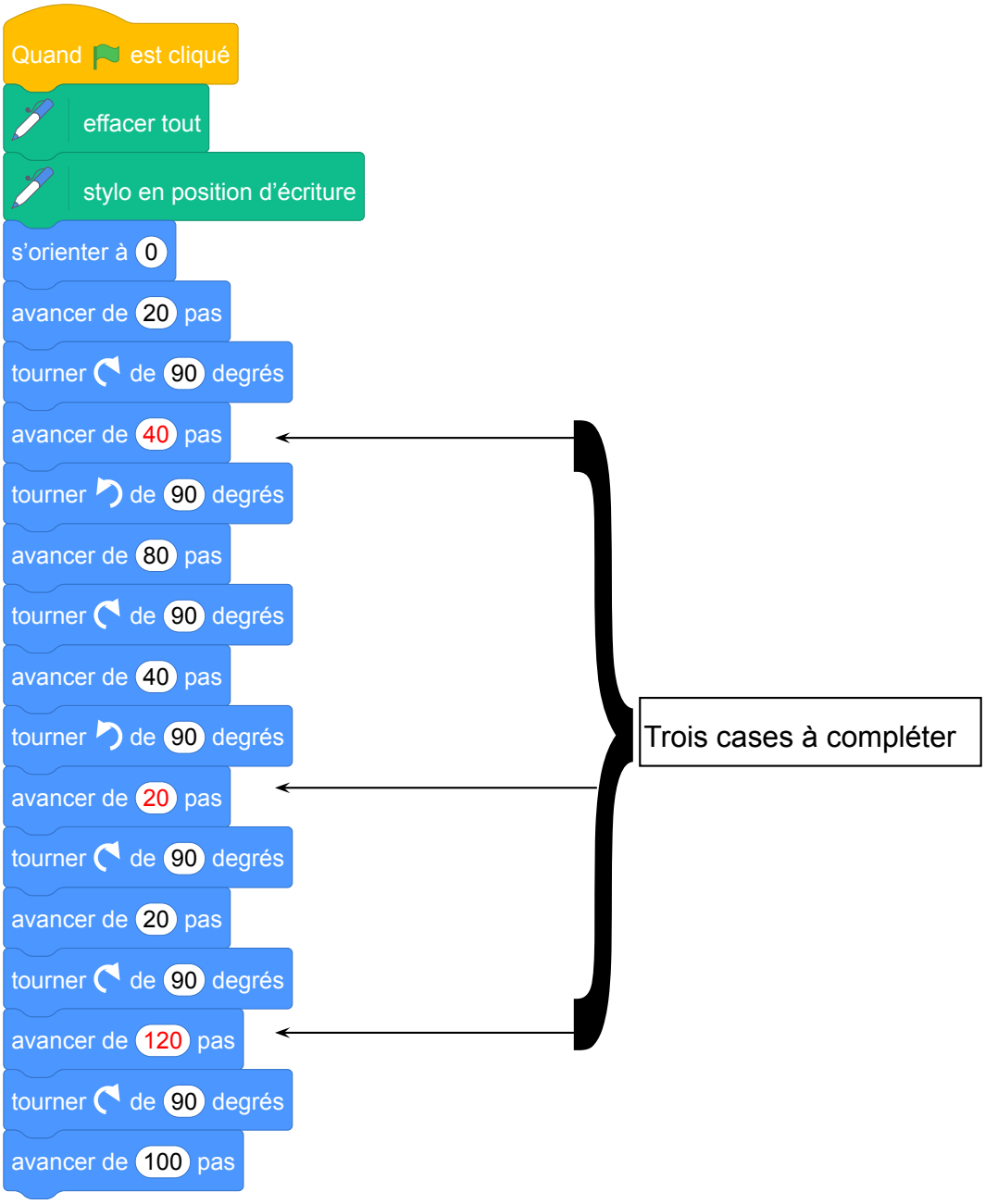
ANNEXE

Exercice 3. Question 1



Chaque côté de carreau mesure 20 pixels.
La position de départ du stylo est indiquée sur la figure ci-contre.

Exercice 3. Question 3



9 Métropole, La Réunion, Antilles-Guyanne, septembre 2022, série générale

Références

Série : générale

Académie : Métropole, La Réunion, Antilles, Guyanne

Date : septembre 2022

Exercice 3 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- déplacement
- grandeurs et mesures
- symétrie

Thèmes informatiques

- activité du lutin

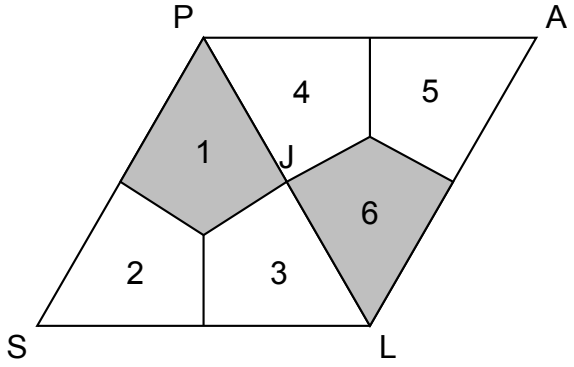
Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	1
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole – La Réunion – Antilles-Guyane, 12 septembre 2022

La figure ci-dessous est un pavage constitué de cerfs-volants.
Les triangles SLP et PLA ainsi formés sont des triangles équilatéraux.

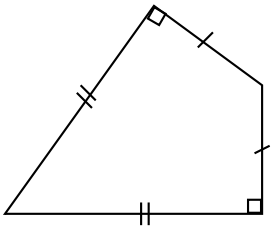


PARTIE A :

1. Déterminer la mesure de l'angle $\widehat{PSL}$.
2. Quelle est l'image du cerf-volant 2 par la symétrie d'axe (PL) ? On ne demande pas de justification.
3. Déterminer par quelle transformation du plan le cerf-volant 1 devient le cerf-volant 6 ?
   On ne demande pas de justification.

PARTIE B :

Dans cette partie, on se propose de construire le cerf-volant ci-dessous.
Essya, Nicolas et Tiago souhaitent construire cette figure à l'aide d'un logiciel de programmation.



Ils écrivent tous un programme « Cerf-volant » différent.

Programme de Essya	Programme de Nicolas	Programme de Tyago
définir Cerf-volant avancer de 300 pas tourner ↺ de 90 degrés avancer de 173 pas tourner ↺ de 60 degrés avancer de 173 pas tourner ↺ de 90 degrés avancer de 300 pas	définir Cerf-volant avancer de 300 pas tourner ↺ de 120 degrés avancer de 300 pas tourner ↺ de 120 degrés avancer de 300 pas	définir Cerf-volant avancer de 173 pas tourner ↺ de 60 degrés avancer de 300 pas tourner ↺ de 90 degrés avancer de 173 pas tourner ↺ de 120 degrés avancer de 300 pas

1. Tracer le programme « Cerf-Volant » de Nicolas, en prenant 1 cm pour 100 pas.
2. Un élève a écrit le script correct. Donner le nom de cet élève en justifiant la réponse.

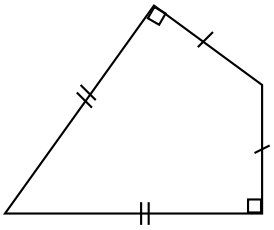
Correction

Fichier scratch : `sg_metropole_2022_septembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/775376214/`

PARTIE A :

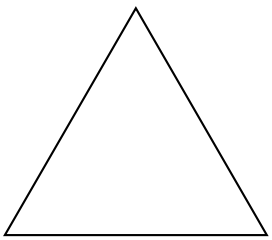
1. Tous les angles d'un triangle équilatéral ont pour mesure $\frac{180}{3} = 60^\circ$ C.
2. L'image du cerf-volant 2 par la symétrie d'axe (PL) est le cerf-volant 5.
3. Le cerf-volant 1 devient le cerf-volant 6 par la symétrie de centre J.

PARTIE B :



Programme de Essya	Programme de Nicolas	Programme de Tyago
définir Cerf-volant avancer de 300 pas tourner de 90 degrés avancer de 173 pas tourner de 60 degrés avancer de 173 pas tourner de 90 degrés avancer de 300 pas	définir Cerf-volant avancer de 300 pas tourner de 120 degrés avancer de 300 pas tourner de 120 degrés avancer de 300 pas	définir Cerf-volant avancer de 173 pas tourner de 60 degrés avancer de 300 pas tourner de 90 degrés avancer de 173 pas tourner de 120 degrés avancer de 300 pas

1. Le programme de Nicolas permet de dessiner un triangle équilatéral de côté 300 pas.



2. Le programme de Tyago ne convient pas car après avoir dessiné un petit côté et tourner de 60° on avance de 300 pas au lieu de 173 pas à nouveau.
   Si on part du petit côté supérieur il faut ensuite tourner à gauche de 90° et non de 60°.
   Ce n'est pas le programme de Nicolas : il ne reste plus que le programme d'Essya.

10 Amérique du Sud, novembre 2022, série générale

Références

Série : générale

Académie : Amérique du Sud

Date : novembre 2022

Exercice 4 - 18 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- rosace
- déplacement
- grandeurs et mesures
- rotation

Thèmes informatiques

- activité du lutin
- itération bornée
- bloc sans paramètre

Compétences travaillées

- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

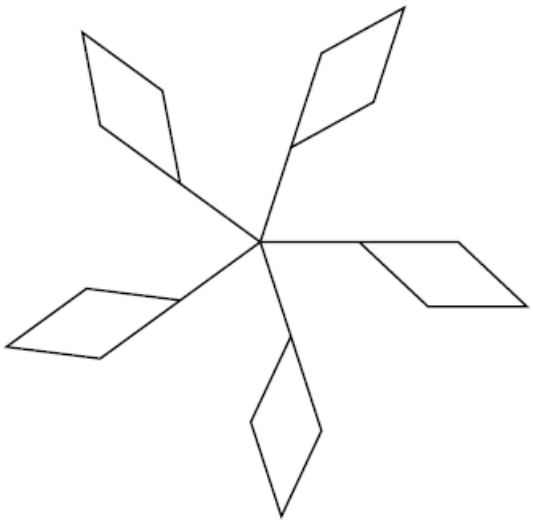
Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Amérique du Sud, novembre 2022

Dans cet exercice. aucune justification n'est attendue

On souhaite réaliser le logo ci-dessous avec le logiciel Scratch à partir du script incomplet ci-dessous.

Logo



Script principal

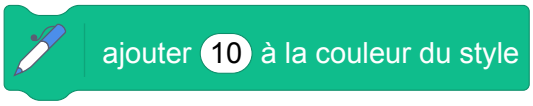
On rappelle que l'instruction s'orienter à 90 consiste à orienter le lutin et le stylo horizontalement vers la droite.

Le bloc Motif permet de réaliser la figure ci-contre :

1. En mathématiques, comment appelle-t-on la transformation géométrique qui permet de passer d'un motif du logo au suivant ?
2. Ici, le stylo est orienté horizontalement vers la droite au départ.
   Parmi les trois propositions suivantes, quelle est celle qui permet d'obtenir le motif souhaité ?

Proposition 1	Proposition 2	Proposition 3
définir Motif stylo en position d'écriture répéter 2 fois avancer de 50 pas tourner de 140 degrés avancer de 50 pas tourner de 40 degrés relever le stylo	définir Motif stylo en position d'écriture répéter 2 fois avancer de 50 pas tourner de 40 degrés avancer de 50 pas tourner de 140 degrés relever le stylo	définir Motif stylo en position d'écriture avancer de 50 pas répéter 2 fois avancer de 50 pas tourner de 40 degrés avancer de 50 pas tourner de 140 degrés relever le stylo

3. Compléter le script principal en recopiant sur la copie uniquement la boucle « répéter» (c'est-à-dire les instructions 4, 5, 6 et 7).

4. On veut placer l'instruction 

de façon à changer de couleur à chaque motif.

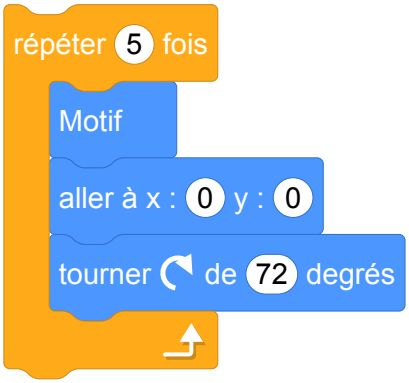
   Sur la copie, indiquer un numéro d'instruction du script principal après laquelle on peut placer cette instruction.

Correction

Fichier Scratch : `sg_amerique_du_sud_2022_novembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/775379772/`

Dans cet exercice aucune justification n'est attendue

1. La transformation est la rotation.
2. C'est la proposition 3.
3. Script complété :



   Rem. on tourne de 72°car $\frac{360}{5} = 72$.

4. On peut placer cette instruction après l'instruction 7.

11 Nouvelle-Calédonie, décembre 2022, série générale

Références

Série : générale

Académie :Nouvelle-Calédonie

Date : décembre 2022

Exercice 8 - 11 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- geometrie
- rosace
- deplacement
- rotation

Thèmes informatiques

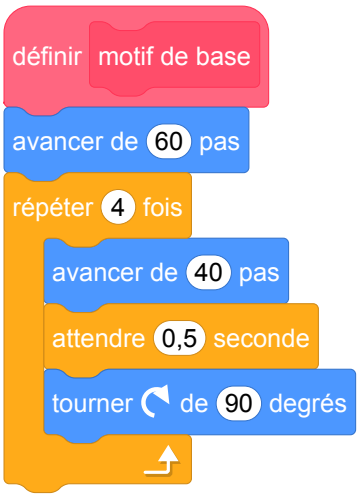
- activite du lutin
- iteration bornee
- bloc sans paramètres

Compétences travaillées

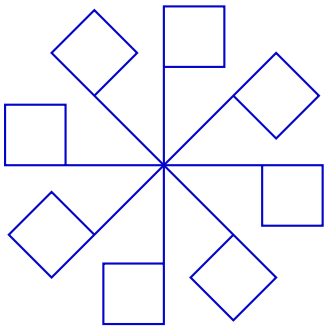
- comprendre un algorithme
- exécuter un algorithme
- compléter un algorithme

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	1
Modification position, orientation etc.	1
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	0
capteur	0
Instructions	
communication	
entrée	0
sortie (affichage avec « dire »)	0
activité du lutin	
déplacement	1
tracé	1
visibilité	0
exécution interne	
variables	
affectation unique	0
changement de valeur	0
chgt avec auto-référence	0
calcul	0
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	1
itération conditionnelle	0
imbrication de structures	1
Procédures	
sans paramètres	1
avec paramètres	0
Gestion du temps	
attente	1
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	0
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

1. Dessiner sur la copie le motif correspondant au script Scratch ci-contre, le stylo étant en position d'écriture.
   On prendra 1 cm pour 10 pas.

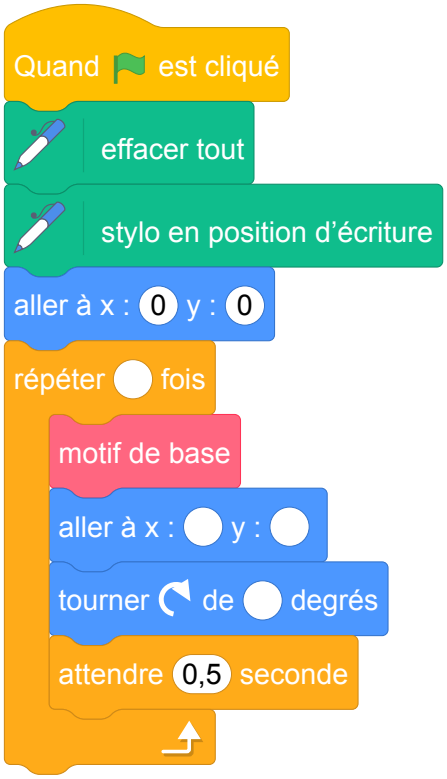


2. **Sur l'annexe**, compléter les informations manquantes du script n° 2 qui permet d'obtenir la figure ci-dessous.



ANNEXE QUESTION 2 À RENDRE AVEC LA COPIE

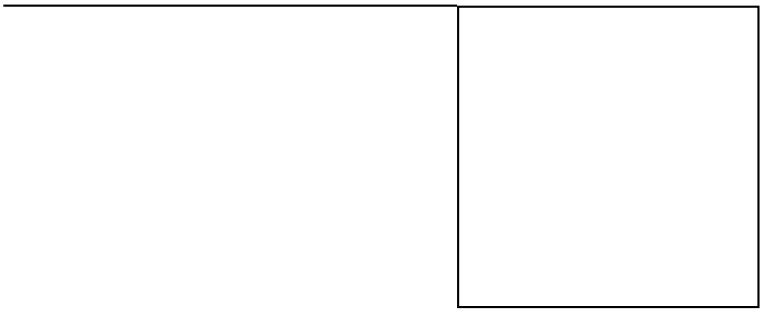
Script n° 2



Correction

Fichier Scratch : `sg_nouvelle_caledonie_2022_decembre.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/791811728/`

1. Motif de base :



2. **Annexe : question 2**

Script n° 2

12 Métropole, juin 2022, série professionnelle

Références

Série : générale

Académie : Métropole-Antilles-Guyane

Date : juin 2022

Exercice 4 - 20 points sur 100

Tags

Origine du document

- brevet

Thèmes mathématiques

- géométrie
- nombres et calculs
- grandeurs et mesures

Thèmes informatiques

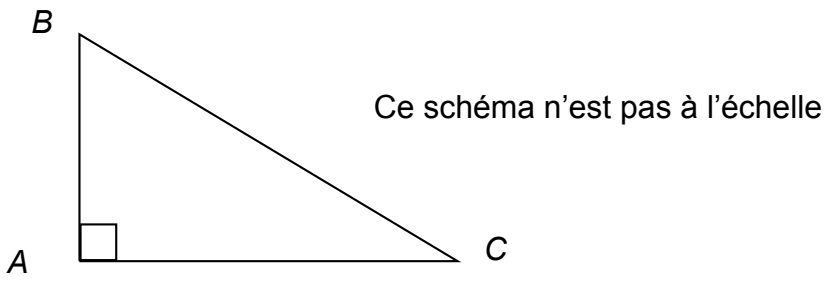
- variable réservée
- variable déclarée
- calcul

Compétences travaillées

- organiser une séquence d'instructions

Attendus	
Objets mobilisés	
spécifiques	
lutin	
état initial	0
Modification position, orientation etc.	0
multiplicité des lutins	0
scène (état du dessin)	0
général	
variables réservées (réponse, x…)	1
variables déclarées	1
capteur	0
Instructions	
communication	
entrée	1
sortie (affichage avec « dire »)	1
activité du lutin	
déplacement	0
tracé	0
visibilité	0
exécution interne	
variables	
affectation unique	1
changement de valeur	0
chgt avec auto-référence	0
calcul	1
aléatoire	0
Structures de contrôle séquentiel	
séquence	0
conditionnelle	
réduite (si-alors)	0
complète (si-alors-sinon)	0
itération bornée	0
itération conditionnelle	0
imbrication de structures	0
Procédures	
sans paramètres	0
avec paramètres	0
Gestion du temps	
attente	0
Structures de contrôle événementiel	
événements extérieurs	0
événements internes	0
messages	0
Expressions	
expression arithmétique	1
expression logique (test de comparaison, d'égalité ...)	0
expression booléenne (ET, OU, NON …)	0
Etat	0

Énoncé - Métropole Antilles-Guyane série professionnelle, 30 juin 2022



1. Parmi les trois propositions suivantes, choisir et recopier la relation qui traduit la propriété de Pythagore appliquée au triangle rectangle ABC représentés ci-dessus.

   On souhaite écrire un programme en langage Scratch permettant de déterminer la longueur de BC connaissant les longueurs AB et AC. Ce programme sera constitué des briques présentées ci-dessous dans le désordre.

   ① mettre BC à racine de AB * AB + AC * AC

   ② mettre AC à Réponse

   ③ demander Quelle est la longueur du côté AB et attendre

   ④ dire regrouper La longueur BC est et BC

   ⑤ demander Quelle est la longueur du côté AC et attendre

   ⑥ quand est cliqué

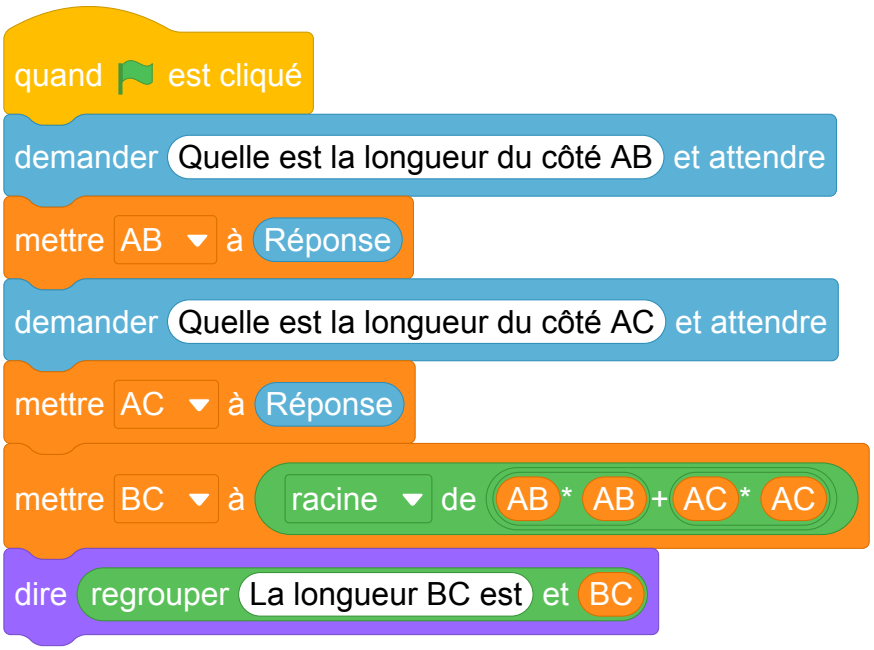
   ⑦ mettre AB à Réponse

2. Ecrire sur votre copie les numéros des briques dans un ordre qui permet de réaliser ce programme.
3. Calculer la longueurs BC si AB = 2,25 cm et AC = 10 cm

Correction

Fichier Scratch : `pro_metropole_2022_juin.sb3`
Lien vers le fichier Scratch sur le Studio Scratch c3i-brevet : `https://scratch.mit.edu/projects/712761536/`

1. $[BC]$ étant l'hypoténuse c'est $BC^2 = AB^2 + AC^2$
2. 6 ; 3 ; 7 ; 5 ; 2 ; 1 ; 4.



3. $BC^2 = 2,25^2 + 10^2$ $BC^2 = 5,0625^2 + 100 = 105,0625$ $BC = \sqrt{100,0625} \simeq 10,003\ cm$

Bibliographie

[1] Sylvie Alayrangues, Emmanuel Beffara, Sebastien Daniel, Christophe Declercq, Anne Heam, Jean-Vincent Loddo, Philippe Marquet, Jean-Christophe Masseron, Antoine Meyer, Malika More, Florence Neny, Vincent Pantaloni, Gaëtan Perrin, Cecile Prouteau, Georges Saliba, Sylviane R. Schwer, Fabien Tarissan, Chloe Ubera, Jean-Marc Vincent, and Emmanuel Volte. Une analyse des exercices d'algorithmique et de programmation du brevet 2017. *Repères IREM*, 116 :47–81, 2019.

[2] APMEP. Association des professeurs de pathématiques de l'pnseignement public. https ://www.apmep.fr/Annales-du-Brevet-des-colleges.

[3] Emmanuel Beffara, Malika More, and Cécile Prouteau. *Algorithmique et programmation au cycle 4 : Commentaires et recommandations du groupe Informatique de la CII Lycée*. Institut de recherche pour l'enseignement des mathématiques, Paris, October 2017.

[4] IREM de Lille. *Scratch au Collège : Algorithme et programmation*. Exo7, 2018.

[5] Anne Heam. La programmation événementielle avec Scratch : moins simple qu'il n'y paraît. *MathemaTICE*, 56, 2017.

[6] Éducation Nationale. Algorithmique et programmation, cycle 4. `https://cache.media.eduscol.education.fr/file/Algorithmique_et_programmation/67/9/RA16_C4_MATH_algorithmique_et_programmation_N.D_551679.pdf`, Mars 2016.

[7] Éducation Nationale. Programme du cycle 4, Novembre 2018. Volet 3 : Les enseignements - Mathématiques (page 146). Texte consolidé à partir du programme BOEN spécial n°11 du 26 Novembre 2015, des nouvelles dispositions publiées au BOEN n°30 du 26 Juillet 2018.

[8] Éducation Nationale. Bulletin officiel spécial n°1 du 22 janvier 2019, Janvier 2019.

Index - Thèmes mathématiques

Agrandissement - Réduction, 66, 80, 120, 126, 138, 142, 166, 263

Calcul sans programme, 43, 102, 107, 142, 166, 198, 238, 242, 246, 249, 254, 263, 288, 291
Caractérisation des figures, 39, 66, 77, 80, 88, 147, 166, 198, 202, 206, 211, 215, 238, 246, 249, 263, 266

Déplacement, 39, 43, 48, 66, 88, 92, 120, 126, 132, 138, 142, 152, 159, 166, 190, 194, 202, 206, 211, 222, 226, 238, 246, 266, 303, 308, 315, 318, 322

Fonctions, 291
Frise, 39, 53, 80, 147, 206, 211, 238, 242, 263, 269, 284, 303

Géométrie, 39, 53, 58, 66, 73, 77, 80, 88, 120, 126, 132, 138, 142, 147, 152, 166, 190, 194, 202, 206, 211, 215, 238, 242, 246, 249, 263, 269, 284, 303, 308, 315, 318, 322, 325
Géométrie sans transformation, 48, 61, 92, 107, 198, 266
Grandeurs et mesures, 53, 73, 77, 80, 107, 120, 142, 147, 159, 166, 190, 194, 202, 211, 215, 231, 238, 242, 246, 249, 266, 269, 297, 315, 318, 325

Homothétie, 66, 138, 142

Inéquations, 291

Nombres et calculs, 25, 32, 35, 98, 102, 107, 166, 176, 231, 238, 246, 249, 254, 257, 260, 263, 280, 288, 291, 294, 300, 325

Pavage, 269
Périmètre, 126, 198, 266
Probabilités, 107, 185, 206, 234, 297
Programme de calcul, 25, 32, 35, 98, 176, 182, 231, 257, 260, 291, 294, 300
Proportionnalité, 43, 77, 102, 202, 234, 242, 269, 297

Repérage, 43, 48, 58, 77, 88, 107, 142, 219, 222, 226, 242, 308
Rosace, 73, 126, 194, 202, 246, 269, 303, 318, 322
Rotation, 73, 202, 215, 246, 269, 303, 318, 322

Simulations, 107, 185, 234
Spirale, 58, 61, 166, 215, 249
Statistiques, 234
Symétrie, 308, 315

Translation, 39, 88, 147, 206, 211, 242, 303, 308

Index - Thèmes informatique

Activité du lutin, 32, 39, 43, 48, 61, 80, 88, 92, 120, 126, 142, 147, 152, 159, 166, 190, 194, 202, 206, 211, 215, 219, 226, 238, 242, 246, 249, 263, 266, 269, 284, 300, 303, 308, 315, 318, 322
Générateur de nombre aléatoire, 107, 166, 185, 206, 234

Bloc avec entrée, 147
Bloc sans entrée, 39, 53, 58, 61, 66, 88, 120, 132, 138, 142, 147, 185, 194, 198, 202, 206, 211, 234, 238, 242, 246, 263, 269, 280, 300, 303, 318, 322
Bloc sans entrée, 43
Branchement conditionel, 25, 92, 107, 166, 172, 185, 206, 219, 234, 254, 288

Calcul, 25, 32, 35, 80, 98, 102, 107, 142, 166, 176, 182, 198, 231, 242, 254, 257, 260, 263, 288, 291, 294, 300, 303, 325
Capteurs, 32, 35, 43, 92, 98, 166, 172, 182, 231
Communication, 25, 32, 92

Etat, 274, 297
Envoi de messages, 98, 172, 176
Programmation événementielle sans message, 25, 32, 43, 219

Boucle bornée, 39, 53, 58, 61, 66, 77, 88, 92, 107, 120, 126, 132, 142, 147, 152, 185, 194, 198, 202, 206, 211, 215, 234, 238, 242, 246, 249, 263, 266, 269, 280, 284, 300, 303, 308, 318, 322
Boucle conditionnelle, 80, 102, 166

Scène, 43, 48, 73, 132, 138, 159, 190, 222, 269

Gestion du temps, 80, 92

Variable déclarée, 32, 35, 58, 61, 66, 80, 92, 98, 102, 107, 120, 138, 142, 152, 166, 172, 176, 182, 185, 190, 194, 215, 231, 234, 249, 257, 260, 263, 291, 294, 300, 308, 325
Variable réservée, 25, 35, 43, 61, 88, 107, 176, 182, 190, 198, 219, 242, 254, 257, 260, 288, 291, 294, 325

Index - Compétences informatiques

Compléter un algorithme, 73, 92, 107, 126, 132, 138, 142, 147, 172, 185, 194, 198, 215, 219, 234, 238, 246, 249, 254, 260, 266, 269, 280, 284, 288, 291, 294, 318, 322
Comprendre un algorithme, 25, 32, 35, 39, 43, 48, 53, 58, 61, 66, 73, 77, 80, 88, 92, 98, 102, 107, 120, 126, 132, 138, 142, 147, 152, 159, 166, 172, 176, 182, 185, 190, 194, 198, 202, 206, 211, 215, 219, 222, 226, 238, 242, 246, 249, 254, 263, 266, 269, 284, 288, 300, 308, 315, 318, 322

Ecrire un algorithme, 73, 132, 185, 215, 274, 297
Exécuter un algorithme, 25, 32, 35, 43, 48, 53, 58, 61, 66, 73, 77, 80, 88, 98, 120, 132, 138, 142, 147, 152, 176, 182, 190, 202, 206, 219, 222, 242, 254, 260, 263, 266, 269, 274, 284, 288, 294, 297, 300, 308, 315, 318, 322

Modifier un algorithme, 39, 58, 61, 77, 126, 152, 159, 166, 182, 190, 202, 206, 222, 242, 263, 284, 300, 308

Organiser une séquence d’instructions, 325

Utiliser une variable avec auto référence, 176, 291