

Algorithmique et Scratch

3^e

Programme du cycle 4 — BO spécial n°11 du 26 novembre 2015, ajusté au BO du 26 juillet 2018

Un programme Scratch au brevet ne se « comprend » pas : **il s'exécute à la main**. On prend son brouillon, on suit les instructions une par une, et on note après chaque ligne la position, la direction et la valeur des variables. Ce tableau s'appelle une **trace**, et c'est la seule technique qui marche — elle marche toujours.

1 Le vocabulaire qui piège

DÉFINITION

Un tableau où l'on écrit **une ligne par instruction exécutée** — pas par bloc écrit — avec une colonne par chose qui change : la position ($x ; y$), la direction, l'état du stylo, la valeur de chaque variable. Une boucle « répéter 4 fois » produit quatre groupes de lignes.

RÈGLE

Les blocs qui se confondent, et ce qu'ils font vraiment. **`aller à x: ... y: ...`** téléporte et ne dessine aucun trajet. **`avancer de`** avance dans la direction actuelle. **`s'orienter à`** remplace la direction — 90 vers la droite, 0 vers le haut, -90 vers la gauche, 180 vers le bas. **`tourner de`** ajoute à la direction actuelle. **`répéter n fois`** exécute son contenu n fois **en tout**.

Le repère de Scratch est celui des mathématiques, pas celui d'un écran : x vers la droite, y vers le haut, centre en $(0 ; 0)$. Un y négatif est en bas. C'est la source du deuxième piège de cette fiche.

2 Reconnaître en dix secondes

RECONNAÎTRE EN DIX SECONDES

- ① **Le signal.** Un script en blocs, et une question du type « quelles sont les coordonnées du point d'arrivée », « quelle figure obtient-on », « que vaut la variable à la fin ».
- ② **Le réflexe qui économise l'exercice.** Repérez la boucle avant tout le reste. Neuf fois sur dix le script trace un polygone régulier, et l'angle donne le nombre de côtés.
- ③ **Le contre-signal.** Pas de boucle et pas de variable : il n'y a rien à deviner, c'est une trace de quatre ou cinq lignes. Faites-la, ne réfléchissez pas.

PROPRIÉTÉ

`répéter n fois [ avancer de  $L$  · tourner de  $a$  ]` trace un **polygone régulier à n côtés** dès que le lutin fait un tour complet :

$$n \times a = 360$$

3 Écrire la trace sans se tromper

RÈGLE

Une ligne par instruction exécutée, pas par bloc écrit, et on recopie la valeur même quand elle ne change pas. Une case vide est une case qu'on relira de travers, et une boucle écrite sur une seule ligne est un décalage garanti.

4 Trois exemples résolus

EXEMPLE

Le lutin part de $(0 ; 0)$, orienté à 90. Script : **`aller à x:-50 y:20`**, **`avancer de 80`**, **`tourner de 90 degrés`**, **`avancer de 40`**. Donner les coordonnées d'arrivée.

- ① **`aller à`** téléporte : la position devient $(-50 ; 20)$ et la direction ne change pas.
- ② Direction 90, donc vers la droite : x augmente de 80, y ne bouge pas.
 $-50 + 80 = 30$
position $(30 ; 20)$
- ③ **`tourner de`** est relatif : on ajoute. Le lutin regarde maintenant vers le bas, et la position n'a pas changé.
 $90 + 90 = 180$
- ④ Direction 180, donc vers le bas : y diminue.
 $20 - 40 = -20$
position $(30 ; -20)$

le lutin arrive en $(30 ; -20)$

EXEMPLE

`stylo en position d'écriture`, puis `répéter 5 fois [ avancer de 70 · tourner de 72 degrés ]`. Quelle figure est tracée ?

- ① Cinq rotations de 72° : le lutin fait un tour complet et revient à sa direction de départ, donc la figure est **fermée**.
 $5 \times 72 = 360$
- ② Chaque passage trace un segment de même longueur et chaque rotation est la même : les cinq côtés et les cinq angles sont égaux.
- ③ Le nombre de côtés se lit directement sur l'angle.
 $360 \div 72 = 5$

un pentagone régulier de côté 70

EXEMPLE

La variable `total` vaut 0 au départ, et `i` vaut 1. Script : `répéter 4 fois [ mettre total à total + i × i · ajouter 1 à i ]`. Que vaut `total` à la fin ?

- ① **Premier passage.** On calcule $i \times i$ avec la valeur **actuelle** de i , puis seulement on incrémente. L'ordre des deux instructions décide de tout.
 $0 + 1 \times 1 = 1$
- ② Deuxième passage, avec $i = 2$.
 $1 + 2 \times 2 = 5$
- ③ Troisième passage, avec $i = 3$.
 $5 + 3 \times 3 = 14$
- ④ Quatrième et dernier, avec $i = 4$. À la sortie i vaut 5, mais la boucle s'arrête : cette valeur n'est jamais utilisée.
 $14 + 4 \times 4 = 30$

total vaut 30

5 Les cinq pièges

PIÈGE

Après `s'orienter à 90` puis `tourner de 90`, noter une direction de 90. — les deux blocs contiennent le même nombre, et on lit le second comme une répétition du premier.

Ce que ça coûte : Toutes les positions suivantes, donc l'exercice entier.

Le réflexe : `s'orienter à` remplace, `tourner de` ajoute. Si le bloc dit « tourner », vous faites une addition — ici $90 + 90 = 180$.

PIÈGE

Après `avancer de 40` vers le bas, noter $y = 60$ au lieu de $y = -20$. — sur un écran, descendre c'est « augmenter » ; dans le repère de Scratch, non.

Ce que ça coûte : La question, et le tracé de la figure s'il y en a un.

Le réflexe : dessinez les deux axes en marge de votre brouillon, avec une flèche vers le haut sur y . Trois secondes, une fois pour toutes.

PIÈGE

`répéter 4 fois` et tracer cinq côtés — ou trois. — on compte les rotations au lieu des avancées, ou l'inverse.

Ce que ça coûte : La figure est fautive d'un côté, donc fautive.

Le réflexe : numérotez les passages 1, 2, 3, 4 dans la marge avant de commencer. On ne compte pas en traçant.

PIÈGE

Tracer un trait entre l'ancienne et la nouvelle position après un `aller à`. — le lutin s'est bien déplacé, donc on croit qu'il a laissé une trace.

Ce que ça coûte : Une figure qui a un segment de trop.

Le réflexe : regardez l'état du stylo dans votre trace, et rappelez-vous que `aller à` téléporte — même stylo baissé, et beaucoup de sujets le précisent.

PIÈGE

Calculer 2×2 au premier passage de l'exemple 3. — on lit les deux lignes de la boucle comme un tout, au lieu de les exécuter dans l'ordre.

Ce que ça coûte : Toutes les valeurs décalées d'un cran, et la réponse finale fautive.

Le réflexe : une ligne par instruction dans la trace, jamais deux instructions sur la même ligne. Le décalage devient visible.

6 Avant de rendre

À VÉRIFIER

- Ai-je fait une **trace écrite**, une ligne par instruction exécutée ?
- Ai-je recopié les valeurs **qui ne changent pas** à chaque ligne ?
- Pour chaque rotation : ai-je **ajouté** (`tourner de`) ou **remplacé** (`s'orienter à`) ?
- Mes y **augmentent-ils vers le haut** ?
- Ai-je numéroté les passages de la boucle avant de commencer ?
- Si c'est un polygone : ai-je vérifié que $n \times \text{angle} = 360$?
- Ai-je **nommé** la figure obtenue, et pas seulement dessinée ?

À RETENIR

- Une **trace** : une ligne par instruction exécutée, et on recopie même ce qui ne change pas.
- `s'orienter à` **remplace** la direction, `tourner de` l'**ajoute**.
- `aller à` **téléporte** : aucun segment tracé, même stylo baissé.
- Dans le repère de Scratch, y augmente vers le **haut**.
- `répéter n fois [ avancer · tourner de a ]` trace un polygone régulier dès que $n \times a = 360$.