



lelivrescolaire.fr

# Livre du professeur - Mathématiques Scratch

Tous les fichiers GeoGebra peuvent être téléchargés sur [LLS.fr/M3Scratch](https://LLS.fr/M3Scratch).

<b>NOTION D'ALGORITHME</b> .....	<b>2</b>
<b>NOTION DE VARIABLES</b> .....	<b>6</b>
<b>NOTION DE BOUCLES</b> .....	<b>10</b>
<b>INSTRUCTIONS CONDITIONNELLES</b> .....	<b>13</b>
<b>DÉCLENCHEMENT D'ACTION PAR UN ÉVÉNEMENT</b> .....	<b>16</b>
<b>SÉQUENCE D'INSTRUCTIONS</b> .....	<b>19</b>
<b>BILAN</b> .....	<b>24</b>

Document sous licence libre Creative Commons





## Notion d'algorithme

### Corrigé exercice 1 :

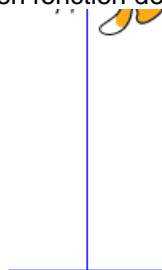
1. On complète le programme comme ci-dessous.



2. Pour que le triangle ait des côtés de longueur 200, il suffit de modifier toutes les instructions « avancer de 100 pas » en « avancer de 200 pas ».

### Corrigé exercice 2 :

Le programme trace une sorte de T orienté en fonction de la position initiale du lutin.



### Corrigé exercice 3 :

Le premier trait tracé mesure  $100 - 10 \times 2 = 80$  pas.

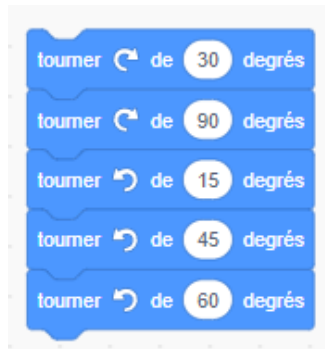
Le second mesure  $(100 - 10) \times 2 = 180$  pas.



Corrigé exercice 4 :



Corrigé exercice A :



Corrigé exercice B :



### Corrigé exercice C :

1.

```
répéter 3 fois  
  aller à position aléatoire  
  jouer le son Miaou jusqu'au bout
```

2.

```
répéter 3 fois  
  aller à position aléatoire  
  jouer le son Miaou jusqu'au bout  
  ajouter taille / 10 à la taille
```

3.

```
répéter 3 fois  
  aller à position aléatoire  
  jouer le son Miaou jusqu'au bout  
  ajouter taille / 10 à la taille  
mettre la taille à 100 % de la taille initiale
```

### Corrigé exercice D :

```
aller à x: 0 y: 0  
s'orienter à 90  
effacer tout  
mettre la taille à 100 % de la taille initiale
```



### Corrigé exercice E :

La suite de commandes suivante permet de représenter le premier dessin.



La suite de commandes suivante permet de représenter le deuxième dessin.





## Notion de variables

### Corrigé exercice 5 :

1. Le programme contient trois variables : a, b et c.
2. On récapitule le contenu de chacune de ces trois variables dans le tableau ci-dessous.

	a	b	c
Ligne 1	5		
Ligne 2	5	3	
Ligne 3	5	3	5
Ligne 4	3	3	5
Ligne 5	3	5	5

3. La variable c permet de conserver en mémoire la valeur d'origine de la variable a.
4. Le programme affiche « Bonjour ! » pendant trois secondes et « Au revoir ! » pendant cinq secondes. Pour afficher la valeur de c, il suffit d'ajouter la ligne ci-dessous au programme.



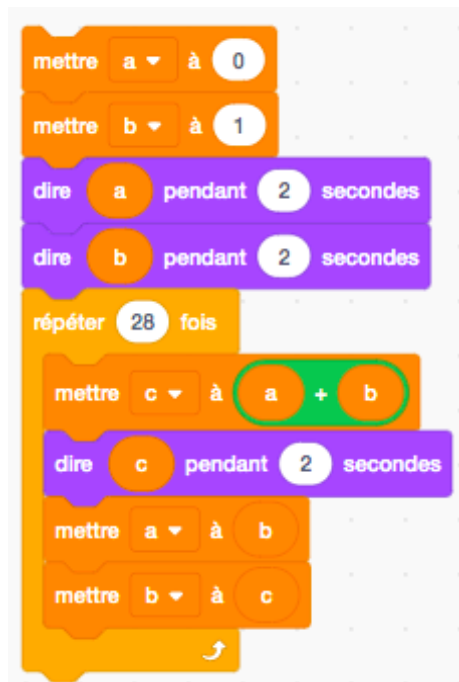
Cette valeur est prévisible : on l'a d'ailleurs prévue lors de la question 2. puisqu'il s'agit de la valeur initiale de a.

### Corrigé exercice 6 :

$$\frac{15 + 12 + 8 + 19 + 11}{5} = 13$$

1. La variable f contient le nombre
2. La variable f correspond à la moyenne des valeurs des variables a, b, c, d et e.

### Corrigé exercice 7 :



Document sous licence libre Creative Commons





### Corrigé exercice A :

Lorsque l'on exécute le programme on obtient 144. Ce résultat est celui de  $12 \times 12$ .

Si l'on souhaite obtenir 132, on doit effectuer l'opération  $11 \times 12$ .

Par exemple, le programme ci-dessous fonctionne.



### Corrigé exercice B :

1. Le premier programme donne *apapa*. Le deuxième donne *papapapa*. Le troisième donne *papaa*.
2. Dans le programme 1, **ma variable** vaut *a* du début à la fin. Dans le programme 2, **ma variable** vaut *a* au début et *papa* à la fin. Dans le programme 3, **ma variable** vaut *a* du début à la fin.

### Corrigé exercice C :

1. La première affirmation est fausse car la variable **c** vaut la somme des variables **a** et **b** qui, au moment où **c** est créée, valent toutes les deux 0. Donc **c = 0**.  
La deuxième affirmation est vraie. Bien que la variable **a** vaille 0 au début du programme, elle est modifiée à l'avant-dernière ligne pour valoir 1.  
La troisième affirmation est fausse. En effet, à la fin du programme, **c = 0** alors que **a + b = 3**.
- 2.





### Corrigé exercice D :

	a	b	c	d
Ligne 1	1			
Ligne 2	1	2		
Ligne 3	1	2	3	
Ligne 4	1	2	3	4
Ligne 5	3	2	3	4
Ligne 6	6	2	3	4
Ligne 7	10	2	3	4
Ligne 8	10	4	3	4
Ligne 9	10	0	3	4

### Corrigé exercice E :

1. Le programme demande le nom de l'utilisateur, puis stocke la réponse dans une variable **nom**. Ensuite, le programme stocke la valeur « Bonjour » dans la variable **politesse**. Enfin, le programme regroupe le contenu des deux variables **politesse** et **nom** et l'affiche.
- 2.

```
demander Quel est ton nom ? et attendre
mettre nom à réponse
mettre politesse à Au revoir
dire regrouper politesse et nom pendant 2 secondes
```

- 3.

```
demander Quel est ton nom ? et attendre
mettre nom à réponse
demander Quel est ton prénom ? et attendre
mettre prénom à réponse
mettre politesse à Bonjour
dire regrouper politesse et regrouper prénom et nom pendant 2 secondes
```





## Corrigé exercice F :

1. Le programme renvoie « Infinity ».
2. La dernière opération devant être effectuée par ce programme est une division par zéro (c'est la valeur prise par la valeur  $x$  à la cinquième ligne du programme). Puisque diviser par zéro est impossible, Scratch renvoie une réponse s'apparentant à un message d'erreur.





## Notion de boucles

### Corrigé exercice 8 :

Le programme répète deux fois l'enchaînement d'instruction « répéter miaou 3 fois puis dire miaou ». En tout, le programme va donc jouer  $2 \times (3 + 1) = 8$  fois le son « Miaou ».

### Corrigé exercice 9 :

Soit  $x$  la valeur à entrer dans la case vide. Le programme va répéter  $2 \times (3 + x) = 6 + 2x$  fois le son « Miaou ». Ainsi, si on souhaite que ce son se joue 14 fois, on doit prendre  $x$  tel que  $6 + 2x = 14$  c'est-à-dire tel que  $2x = 8$ . On doit donc entrer dans la case vide la valeur 4.

### Corrigé exercice 10 :



### Corrigé exercice 11 :

- 1.
2. Si l'utilisateur donne une mauvaise réponse à la question, le programme la repose tant que la bonne réponse n'est pas donnée.
3. Quand l'utilisateur donne une bonne réponse, le programme peut enfin sortir de la boucle « Tant que » et il joue le son « Miaou ».

### Corrigé exercice A :

1. La plus grande ordonnée sera atteinte quand le test **ordonnée y > 50** sera vrai. C'est-à-dire quand l'ordonnée sera égale à 53.
2. La plus grande ordonnée sera atteinte quand le test **ordonnée y > 50** sera vrai. C'est-à-dire quand l'ordonnée sera égale à 58.

### Corrigé exercice B :

1. On entendra 21 fois le son « Miaou » puisqu'on répètera 7 fois l'action d'entendre 3 fois le son « Miaou ».
2. Si on supprime la deuxième brique violette, on entendra 14 fois le son « Miaou ». Si on supprime la première brique violette, on entendra 7 fois le son « Miaou ».
3. En répétant 17 fois la boucle principale et 0 fois la boucle intérieure, il est possible de répéter 17 fois le son « Miaou ». Si on interdit l'utilisation d'une boucle se répétant seulement une fois, cette tâche devient impossible.
4. En répétant 1 fois la boucle principale et 0 fois la boucle intérieure, il est possible de jouer une seule fois le son « Miaou ».



### Corrigé exercice C :

1. L'instruction a été exécutée  $2^5 = 32$  fois.
2. L'instruction n'est pas dans une boucle, elle n'est exécutée qu'une seule fois.
3. L'ordonnée finale du lutin est de  $-200 + 32 \times 5 = -40$ . L'abscisse n'a pas été modifiée par les boucles, elle reste donc à 0.

### Corrigé exercice D :

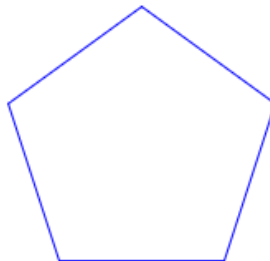
Les deux programmes suivants ont le même effet.



Dans ces deux programmes, le lutin avance de 60 pas.  
Les deux autres programmes font avancer le lutin respectivement de 10 et de 3 600 pas.

### Corrigé exercice E :

1. La figure tracée est la suivante.



2. Pour tracer un carré il faut répéter 4 fois : avancer de 100 puis tourner de 90.



3. Il suffit de répéter 5 fois les instructions de la boucle





## Instructions conditionnelles

### Corrigé exercice 12 :

La valeur de la variable  $a$  est initialisée à 0. On répète ensuite 10 fois l'instruction : « Dire Bonjour si  $a < 4$ , penser Hmm... et ajouter 1 à  $a$  sinon. ». Comme  $a = 0 < 4$ , on est dans le premier cas. Donc, à chaque nouvelle répétition de la boucle, le programme affichera « Bonjour ! ». Le programme affichera donc 10 fois le texte « Bonjour ! ».

### Corrigé exercice 13 :



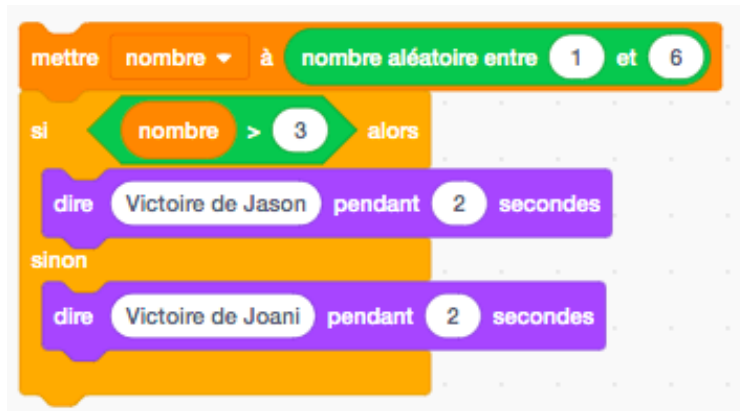
### Corrigé exercice 14 :

Le programme demande un mot à l'utilisateur et affiche « Le mot commence par la lettre a » si la première lettre du mot est un « a ».

### Corrigé exercice 15 :

Le programme de Jason affiche bien un nombre aléatoire entre 1 et 6, mais uniquement si ce nombre est supérieur à 4. Dans le cas contraire, le programme affiche 4. Ainsi Jason est sûr de toujours gagner.

### Corrigé exercice 16 :





### Corrigé exercice A :

1. Le lutin ne dit pas bonjour à chaque exécution du programme, cela dépend de sa.
2. Pour que le lutin dise « Bonjour ! » il faut que son abscisse soit strictement négative.
- 3.



### Corrigé exercice B :

1. Ces deux programmes ne sont pas équivalents. Par exemple si le lutin se trouve en  $(1; 2)$ , le premier programme jouera deux fois le son « Miaou » (une fois car l'ordonnée est strictement positive et une fois car l'abscisse est strictement positive) alors que le programme 2 ne jouera qu'une seule fois ce son.
2. Le programme 1 jouera deux fois le son « Miaou », le programme 2 le jouera deux fois.

### Corrigé exercice C :

- 1.



- 2.





### Corrigé exercice D :

1. Il faut entrer un nombre strictement inférieur à 50.
- 2.



### Corrigé exercice E :

1. Si le lutin a une abscisse positive alors l'abscisse du lutin est remise à 0, tout en gardant à la même ordonnée.
- 2.





## Déclenchement d'action par un événement

### Corrigé exercice 17 :

Ce programme est déclenché par l'appui sur la touche espace.

### Corrigé exercice 18 :

1. Les instructions exécutées lorsqu'on appuie sur la touche a sont les suivantes.



2. Si on appuie cinq fois sur la touche a, le son monte 5 fois de 30. Il faut donc le descendre de 150, c'est-à-dire appuyer trois fois sur la touche b.

### Corrigé exercice 19 :

1. Rachid doit appuyer sur la touche a pour exécuter ce programme.
2. Rachid doit appuyer sur la touche « Flèche du haut » pour arrêter ce programme.

### Corrigé exercice 20 :



Document sous licence libre Creative Commons





### Corrigé exercice 21 :



### Corrigé exercice A :

1. Chaque fois que l'on presse sur la touche espace, le lutin avance de 100 pas et tourne de 90 degrés. Ainsi, si on appuie quatre fois sur la touche espace, le lutin trace un carré.
- 2.



- 3.



### Corrigé exercice B :

1. Le programme est mis en marche en cliquant sur le lutin.
2. Si le lutin se trouve aux coordonnées (20 ; 30) alors le lutin se déplace à des coordonnées aléatoires. Si le lutin se trouve aux coordonnées (-50 ; -100) le lutin ne bouge pas mais dit « Réussi ».
3. La différence est que lorsque le lutin est cliqué et dit « Réussi » le programme s'arrête.

### Corrigé exercice C :

Le programme joue  $4 \times (2 \times (1 + 3) + 3) = 44$  fois le son « Miaou ».

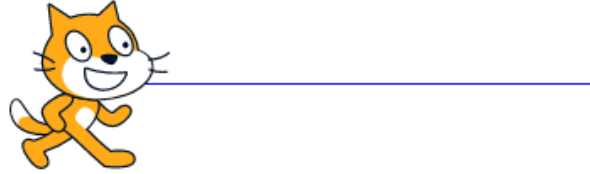
Remarque : les temps d'attente sont nécessaires pour éviter que les sons se superposent.





### Corrigé exercice D :

1. Le lutin trace un trait de 300, se retourne de  $180^\circ$ , revient à son point de départ puis s'oriente vers la droite, comme initialement.



- La séquence *abba* tracera un trait de 100 avant de faire revenir le lutin sur ses pas.
2. La séquence *aabaabaaba* tracera un carré de côté 200.

### Corrigé exercice E :

1. Le programme est lancé par l'appui sur la touche espace.
2. Avec le programme de base, on obtient la figure suivante.



En inversant les briques bleues, on obtient la figure suivante.



Dans le premier cas, le lutin avance avant de tourner. Dans le deuxième cas, le lutin tourne puis avance.

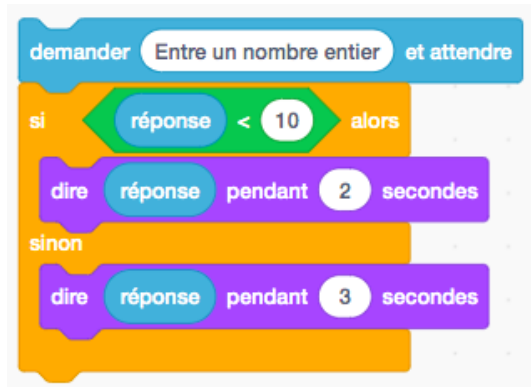


## Séquence d'instructions

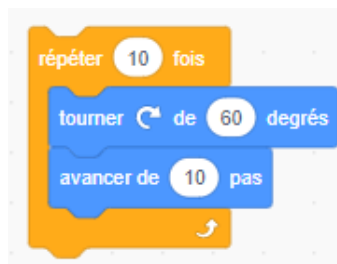
Corrigé exercice 22 :

- « aaa » donne à la variable x la valeur 8 ce qui est insuffisant pour arrêter le programme ;
- « aac » donne à la variable x la valeur 7 ce qui est insuffisant pour arrêter le programme ;
- « cccc » donne à la variable x la valeur 12 ce qui est suffisant pour arrêter le programme ;
- « cacc » donne à la variable x la valeur 11 ce qui est suffisant pour arrêter le programme.

Corrigé exercice 23 :



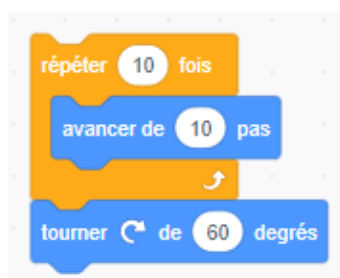
Corrigé exercice 24 :



Ce programme répète 10 fois l'enchaînement des deux opérations « tourner » et « avancer ».



Ce programme commence par tourner puis répète 10 fois l'instruction « avancer ».



Ce programme commence par répéter 10 fois l'instruction « avancer » puis tourne.





### Corrigé exercice 25 :

Les programmes B et C sont équivalents au programme de l'énoncé.

### Corrigé exercice A :



### Corrigé exercice B :

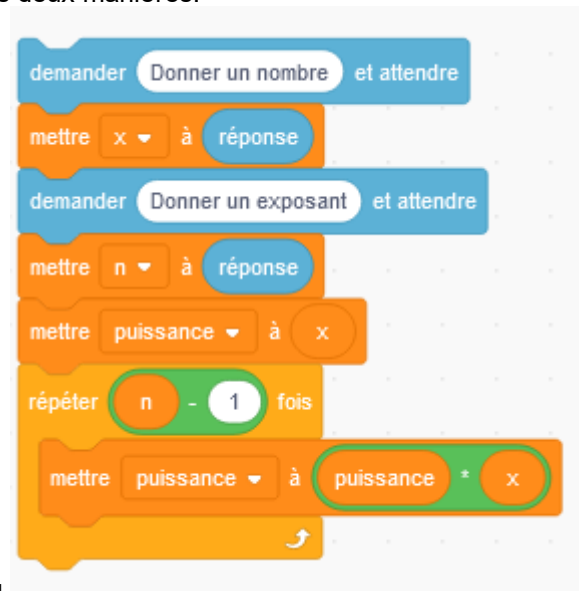
Choisir un nombre.  
Ajouter 7.  
Multiplier le résultat obtenu par lui-même.  
Ajouter 1 au résultat.  
Multiplier le résultat par le nombre de départ.

### Corrigé exercice C :

1. Le programme affiche 16, ce qui n'est pas la bonne réponse. Le programme de Teddy ne fonctionne pas.
2. On peut modifier le programme d'une de ces deux manières.



ou



3.





```
demander Donner un nombre et attendre
mettre x à réponse
demander Donner un exposant entier positif et attendre
mettre n à réponse
si n < 0 alors
  dire Le programme n'est pas prévu pour les exposants négatifs. pendant 2 secondes
  stop tout
mettre puissance à 1
répéter n fois
  mettre puissance à puissance * x
```





### Corrigé exercice D :

1.

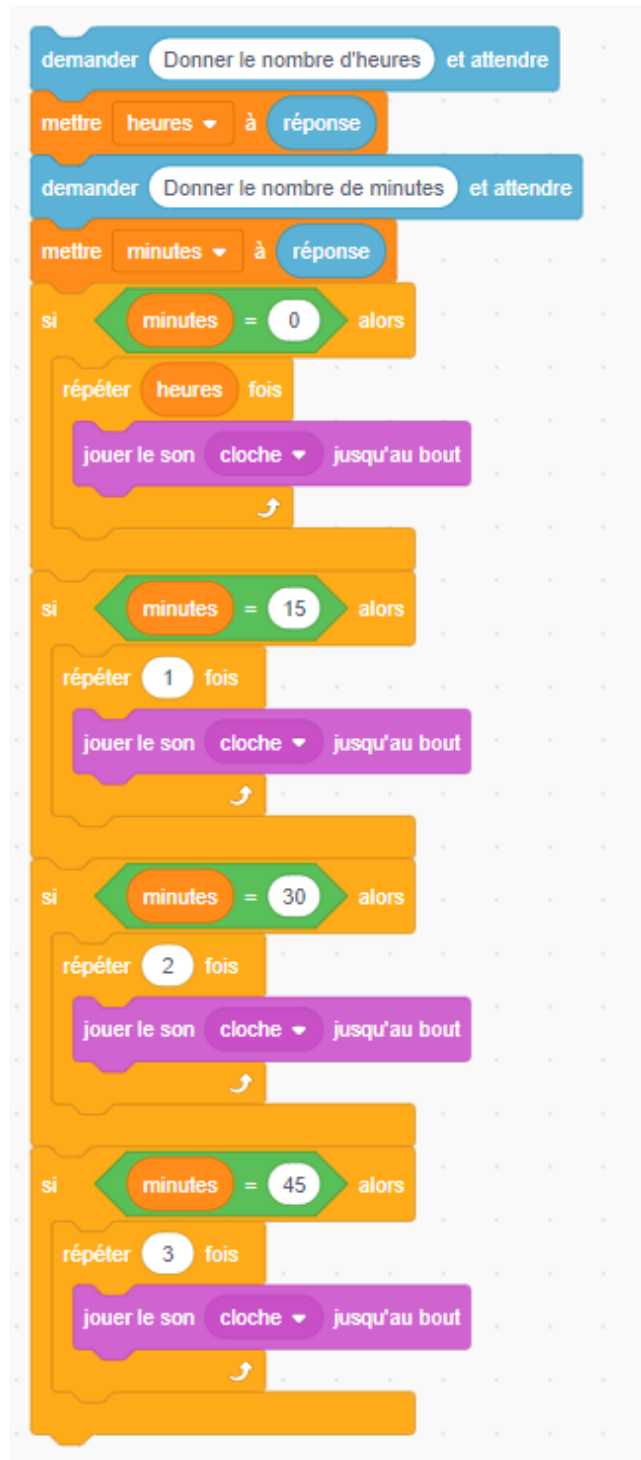
```
répéter 8 fois
  répéter 4 fois
    avancer de 20 pas
    tourner de 90 degrés
  avancer de 20 pas
```

2.

```
relever le stylo
aller à x: 0 y: 0
stylo en position d'écriture
mettre k à 0
répéter 8 fois
  répéter 8 - k fois
    répéter 4 fois
      avancer de 20 pas
      tourner de 90 degrés
    avancer de 20 pas
  ajouter 1 à k
  relever le stylo
  aller à x: 0 y: k * 20
  stylo en position d'écriture
```



Corrigé exercice E :



```
demander Donner le nombre d'heures et attendre
mettre heures à réponse
demander Donner le nombre de minutes et attendre
mettre minutes à réponse
si minutes = 0 alors
  répéter heures fois
    jouer le son cloche jusqu'au bout
si minutes = 15 alors
  répéter 1 fois
    jouer le son cloche jusqu'au bout
si minutes = 30 alors
  répéter 2 fois
    jouer le son cloche jusqu'au bout
si minutes = 45 alors
  répéter 3 fois
    jouer le son cloche jusqu'au bout
```

The image shows a Scratch script designed to play a bell sound based on user input. It starts by asking for the number of hours and minutes, storing the answers in variables. Then, it uses conditional logic to play the bell sound a specific number of times: once for every hour, plus additional times for 15, 30, and 45 minutes. For example, if the user enters 2 hours and 30 minutes, the bell will be played 5 times (2 for hours + 2 for minutes).



## Bilan

### Corrigé exercice A :

1. Pour que le lutin « tombe » il faut que son ordonnée baisse. Ainsi le bon programme est le programme n°1.
- 2.



- 3.



### Corrigé exercice B :

1. Le dessin tracé est une ligne pointillée de 200 pas de long au total et construit de la façon suivante : 10 pas pleins suivis de 10 pas blancs.
- 2.





### Corrigé exercice C :

1. Le programme affiche que 7 et 19 sont des nombres premiers mais que 9 et 15 ne le sont pas. Le programme semble fonctionner normalement.
- 2.

```
demander donner un nombre entier supérieur à 0 et attendre
si réponse = 1 alors
  dire Ce n'est pas un nombre premier pendant 2 secondes
  stop tout
sinon
  si réponse = 2 alors
    dire C'est un nombre premier pendant 2 secondes
    stop tout
  sinon
    mettre ma variable à 2
    répéter jusqu'à ce que ma variable = réponse - 1
    si réponse modulo ma variable = 0 alors
      dire Ce n'est pas un nombre premier pendant 2 secondes
      stop tout
    mettre ma variable à ma variable + 1
  dire C'est un nombre premier pendant 2 secondes
  stop tout
```

3. On ajoute ce bloc en début de programme.





```
demander donner un nombre entier supérieur à 0 et attendre
si réponse < 0 alors
  répéter jusqu'à ce que réponse > 0
  demander donner un nombre entier supérieur à 0 et attendre
si réponse = 1 alors
```

### Corrigé exercice D :

1. C'est la variable  $n$ . En effet, c'est elle qui contrôle le nombre de répétitions de la boucle.
2.  $x$  représente la longueur des traits tracés.
3. Pour la spirale carrée, on modifie le programme comme suit.

```
stylo en position d'écriture
mettre n à 20
mettre x à 0
répéter n fois
  avancer de x pas
  tourner de 90 degrés
  mettre x à x + 5
cacher
```

Et pour la spirale triangulaire, comme le suivant.

```
stylo en position d'écriture
mettre n à 8
mettre x à 0
répéter n fois
  avancer de x pas
  tourner de 120 degrés
  mettre x à x + 10
cacher
```





### Corrigé exercice E :

1.

```
répéter indéfiniment
  aller à x: abscisse x + 1 y: ordonnée y
  si touche flèche haut pressée ? alors
    s'orienter à 0
  si touche flèche bas pressée ? alors
    s'orienter à 180
  si touche flèche droite pressée ? alors
    s'orienter à 90
  si touche flèche gauche pressée ? alors
    s'orienter à -90
```

2.

```
répéter indéfiniment
  répéter jusqu'à ce que touche le bord ?
  aller à x: abscisse x + 1 y: ordonnée y
  si touche flèche haut pressée ? alors
    s'orienter à 0
  si touche flèche bas pressée ? alors
    s'orienter à 180
  si touche flèche droite pressée ? alors
    s'orienter à 90
  si touche flèche gauche pressée ? alors
    s'orienter à -90
```



3.

