

Mon livret d'algorithmique

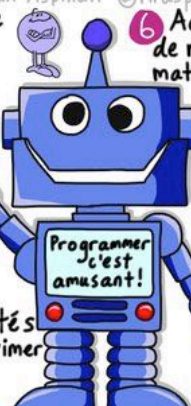
Nom :

Prénom :

Classe :



10 Raisons d'enseigner la Programmation
Par Brian Aspinall @mraspinall
Traduit en français par @flocquet



- 1 Permet de créer du contenu et pas juste d'en consommer
- 2 Développe des capacités et entraîne à s'exprimer autrement
- 3 Apprend à raconter des histoires avec des jeux et des animations
- 4 Pousse les élèves à prendre des risques sans conséquences graves
- 5 Développe la confiance en soi et permet l'inclusion de chacun
- 6 Accompagne et illustre de nombreux principes mathématiques et de grammaire
- 7 Apprend à résoudre des problèmes et mobilise une pensée critique & analytique
- 8 Fait partie de la nouvelle littérature et sera utile pour de nombreux métiers
- 9 Développe des compétences de travail d'équipe et de coopération
- 10 Savoir programmer peut aider l'humanité

BONUS: Programmer vous donne des super pouvoirs

© sylvia duckworth

SOMMAIRE

SEQUENCE N°1 - LES INSTRUCTIONS.....	3
• Je découvre et j'applique	
• Niveau 1	
SEQUENCE N°2 - LES BOUCLES.....	8
• Je découvre et j'applique	
• Niveau 1	
SEQUENCE N°3 - LES VARIABLES.....	12
• Je découvre et j'applique	
• Niveau 1	
SEQUENCE N°4 - LES INSTRUCTIONS CONDITIONNELLES.....	16
• Je découvre et j'applique	
• Niveau 1	
SEQUENCE N°5 - LES BLOCS D'INSTRUCTIONS.....	20
• Je découvre et j'applique	
• Niveau 1	
J'EVALUE MON TRAVAIL.....	24

Séquence N°1

-

Les instructions





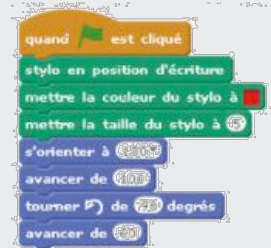
Je découvre

Une **séquence d'instructions** est une suite d'actions à exécuter dans un ordre donné.

Exemple 1 : Pour dessiner la lettre **L**, on peut exécuter les instructions suivantes.

- Prendre un crayon rouge
- Tracer un trait vertical vers le bas de longueur 1 cm
- Tracer un trait horizontal vers la droite de longueur 0,6 cm

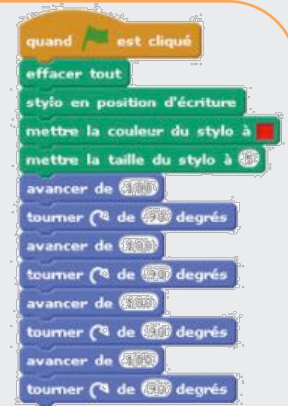
Dans le logiciel Scratch, la **séquence d'instructions ci-contre va dessiner la lettre L à l'écran.**



Exemple 2 : Pour tracer un carré rouge, on peut exécuter les instructions suivantes.

- Prendre un crayon rouge
- Tracer un trait vertical vers le bas de longueur 5 cm
- Tracer un trait horizontal vers la droite de longueur 5 cm
- Tracer un trait vertical vers le haut de longueur 5 cm
- Tracer un trait horizontal vers la gauche de longueur 5 cm

Dans Scratch, l'**algorithme ci-contre permettra de tracer un carré rouge à l'écran.**



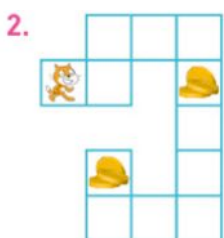
J'applique

Exercice 1 : Le labyrinthe

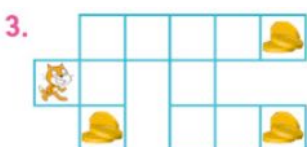
Dans chaque cas, aider Scratchy à récupérer le maximum de pièces d'or en utilisant les instructions , , et pour se déplacer.



Instructions :
.....
.....



Instructions :
.....
.....



Instructions :
.....
.....

Exercice 2 : À l'eau les mains !



Voici un algorithme

Ouvrir le robinet
Se remonter les manches
Mouiller ses mains
Mettre du savon dans ses mains
Frotter longtemps
Rincer ses mains
Sécher ses mains avec une serviette
Fermer le robinet

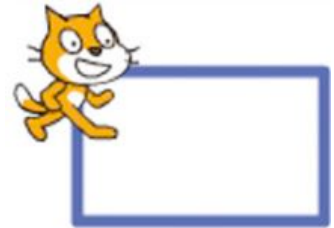
1. Expliquer brièvement ce que permet de faire cet algorithme.
.....
.....

2. Écrire un algorithme qui fasse la même chose, mais qui soit plus respectueux de la planète en économisant au maximum l'eau utilisée.
.....
.....
.....
.....
.....



Exercice 3 : Je choisis le bon programme

1. Parmi les trois programmes suivants, cocher celui qui permet d'obtenir un rectangle comme représenté ci-contre dans le logiciel Scratch.
(On considère que le stylo est déjà en position d'écriture, en bleu, de taille 5.)



Programme ①

```

quand [drapeau] est cliqué
  avancer de 100
  tourner [90] de 90 degrés
  avancer de 60
  tourner [90] de 90 degrés
  avancer de 100
  tourner [90] de 90 degrés
  avancer de 60
  tourner [90] de 90 degrés
  
```

Programme ②

```

quand [drapeau] est cliqué
  avancer de 100
  tourner [60] de 60 degrés
  avancer de 90
  tourner [60] de 60 degrés
  avancer de 100
  tourner [60] de 60 degrés
  avancer de 90
  tourner [60] de 60 degrés
  
```

Programme ③

```

quand [drapeau] est cliqué
  avancer de 100
  tourner [100] de 100 degrés
  avancer de 100
  tourner [100] de 100 degrés
  avancer de 100
  tourner [100] de 100 degrés
  avancer de 100
  tourner [100] de 100 degrés
  
```

2. Expliquer brièvement pourquoi les deux autres programmes ne permettent pas de dessiner un rectangle.

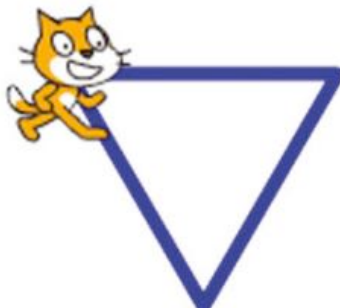
.....

.....

.....

Exercice 4 : Je complète un programme

Compléter les zones blanches dans le programme ci-contre afin que celui-ci permette d'obtenir un triangle équilatéral de côté 100 pixels.



```

quand [drapeau] est cliqué
  effacer tout
  stylo en position d'écriture
  avancer de .....
  tourner [ ] de ..... degrés
  avancer de .....
  tourner [ ] de ..... degrés
  avancer de .....
  tourner [ ] de ..... degrés
  
```



Exercice 5 : J'ordonne un programme

Remettre dans l'ordre les instructions ci-contre afin de réaliser un programme dans Scratch qui permette d'obtenir la lettre M.

- quand est cliqué
- stylo en position d'écriture
- avancer de 50
- tourner de 140 degrés
- mettre la taille du stylo à 5
- avancer de 100
- tourner de 100 degrés
- effacer tout
- s'orienter à 0
- avancer de 50
- tourner de 140 degrés
- aller à x: 0 y: 0
- avancer de 100

Instructions :

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....



Exercice 6 : Je complète un programme à structure imposée

Compléter le programme ci-contre afin d'obtenir le dessin de marche d'escalier ci-dessous où chaque segment mesure 100 pixels. Pour cela, on pourra tester les différentes possibilités avec Scratch jusqu'à obtenir le programme désiré en respectant les instructions utilisées.



.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Séquence N°2

-

Les boucles

*Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison
Je ne contredirai pas le prof de maths, il a toujours raison*





Je découvre

Une **boucle** permet de faire répéter la même action un certain nombre de fois, ou bien jusqu'à ce qu'un événement se produise, ou encore indéfiniment.

Exemple 1 : Pour monter un escalier de 10 marches, on pourrait utiliser cette suite d'instruction :

- Monter une marche
- Monter une marche
- Monter une marche
- Monter une marche
- Monter une marche
- Monter une marche
- Monter une marche
- Monter une marche
- Monter une marche
- Monter une marche

Que l'on peut remplacer par :

Répéter 10 fois
| Monter une marche

ou encore :

Répéter jusqu'à ce que l'on soit en haut de l'escalier
| Monter une marche



Exemple 2 : Pour programmer le clignotement d'une guirlande de Noël.

Répéter indéfiniment

- Allumer les ampoules
- Attendre 1 seconde
- Éteindre les ampoules
- Attendre 1 seconde



Coup de pouce

- Les boucles permettent donc de réduire la taille des programmes et des algorithmes que l'on crée.
- Les boucles Répéter indéfiniment permettent de programmer des actions permanentes.

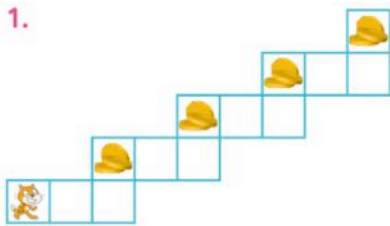


J'applique

Exercice 1 : Le labyrinthe

Dans chaque cas, aider Scratchy à récupérer le maximum de pièces d'or en utilisant les instructions et pour se déplacer à l'intérieur d'une boucle.

1.



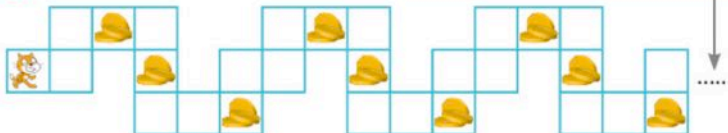
Ces pointillés « » signifient que l'on continue de la même manière un certain nombre de fois.

2.



Ces pointillés « » signifient que l'on continue de la même manière indéfiniment.

3.



Instructions N°1 :

.....

.....

.....

Instructions N°2 :

.....

.....

.....

Instructions N°3 :

.....

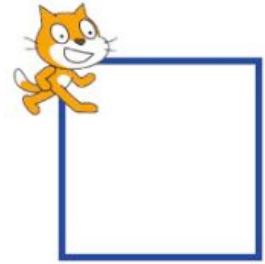
.....

.....



Exercice 2 : Je choisis le bon programme

1. Parmi les trois programmes suivants, cocher celui qui permet d'obtenir le dessin ci-contre dans le logiciel Scratch.



Programme ①

```

quand cliqué
effacer tout
stylo en position d'écriture
répéter 4 fois
  avancer de 100
  tourner de 90 degrés
relever le stylo
    
```

Programme ②

```

quand cliqué
effacer tout
stylo en position d'écriture
répéter 4 fois
  avancer de 100
  tourner de 90 degrés
relever le stylo
    
```

Programme ③

```

quand cliqué
effacer tout
stylo en position d'écriture
répéter 4 fois
  tourner de 90 degrés
  avancer de 100
relever le stylo
    
```

2. Expliquer brièvement pourquoi les deux autres programmes ne permettent pas de dessiner un carré.

.....

.....

.....

.....

.....

Exercice 3 : Je complète un programme

Compléter les zones blanches dans le programme ci-contre afin que celui-ci permette d'obtenir un rectangle de longueur 150 pixels et de largeur 60 pixels.



```

quand est cliqué
effacer tout
stylo en position d'écriture
répéter ..... fois
  avancer de .....
  tourner de ..... degrés
  avancer de .....
  tourner de ..... degrés
    
```



Exercice 4 : J'ordonne un programme

Remettre dans l'ordre les instructions ci-dessous afin de réaliser un programme dans Scratch qui permette d'obtenir un losange de longueur 100 pixels.

stylo en position d'écriture

répéter 2 fois

avancer de 100

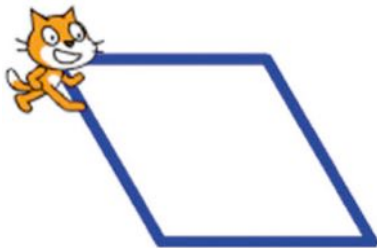
tourner de 60 degrés

avancer de 100

tourner de 120 degrés

quand est cliqué

effacer tout



Instructions :

.....

.....

.....

.....

.....

.....

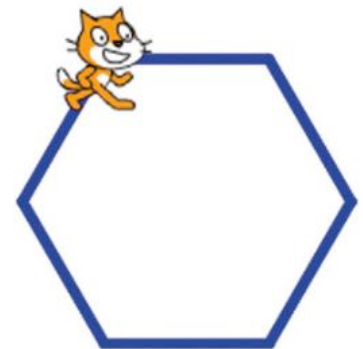
.....

.....

.....

Exercice 5 : Je complète un programme à structure imposée

Compléter le programme ci-dessous afin d'obtenir un hexagone de côté 80 pixels comme le montre le dessin ci-contre. Pour cela, tester à l'aide du logiciel Scratch jusqu'à obtenir le programme désiré en respectant les instructions utilisées.



.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Séquence N°3

-

Les variables





Je découvre

Une **variable** est un espace de stockage dans laquelle on range une valeur (un nombre, un texte) pour l'utiliser dans un programme.
Chaque variable est désignée avec un nom qui doit être défini avant l'écriture du programme.

Exemple : Dans un jeu vidéo, les variables **score**, **nombre de vies** peuvent évoluer le long d'une partie :

SCORE 150

Nombre de vies 3



Dans Scratch, on peut Créer une variable à partir du menu **Données** : on commence par donner un nom à la **variable** pour ensuite l'utiliser dans des actions.

Nouvelle variable

Nom de la variable: SCORE

Pour tous les lutins Pour ce lutin uniquement

Ok Annuler

mettre SCORE à 0

ajouter à SCORE 1

montrer la variable SCORE

cacher la variable SCORE

Coup de pouce

Une **variable** par défaut existe sur Scratch : la variable **réponse**

qui enregistre la réponse de l'utilisateur à une question posée comme demander Donne un nombre et attendre



J'applique

Exercice 1 : Le tir à l'arc

Un archer tire 5 flèches sur une cible avec des zones valant de 1 à 10 points. Le nombre total de points du joueur (**score total**) augmente donc à chaque flèche en ajoutant la valeur du nouveau tir (**valeur fleche**).
À l'aide d'une boucle et de deux variables **score** et **valeur fleche**, compléter l'algorithme décrivant le déroulement d'une épreuve de tir à l'arc et permettant de calculer au fur et à mesure le score du tireur, puis d'annoncer son score final.

Instructions :

Mettre **score total** à

Répéter fois

Annoncer



Exercice 2 : L'horloge digitale

Pour faire fonctionner une horloge digitale, il y a trois variables : **heures**, **minutes** et **secondes**.

Ci-contre, un morceau de l'algorithme permettant de gérer l'affichage de l'heure (de 0 à 24 heures).

Répéter 60 fois
Afficher secondes
Ajouter 1 à secondes
Mettre secondes à 0
Ajouter 1 à minutes

Compléter cet algorithme pour qu'il permette d'afficher les heures, minutes et secondes sur une journée complète.

Instructions :

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....



Exercice 3 : Je choisis le bon programme

1. Parmi les trois programmes suivants, cocher celui qui permet de calculer le triple d'un nombre donné.

Programme ①

```

quand [drapeau] est cliqué
demander Donner un nombre. et attendre
mettre nombre à réponse
dire regroupe Son triple est nombre * 2
    
```

Programme ②

```

quand [drapeau] est cliqué
demander Donner un nombre. et attendre
mettre nombre à réponse
dire regroupe Son triple est nombre * 3
    
```

Programme ③

```

quand [drapeau] est cliqué
demander Donner un nombre. et attendre
mettre nombre à réponse
dire regroupe Son triple est nombre / 3
    
```

2. Expliquer pourquoi les autres programmes ne permettent pas de calculer le triple d'un nombre donné.

.....

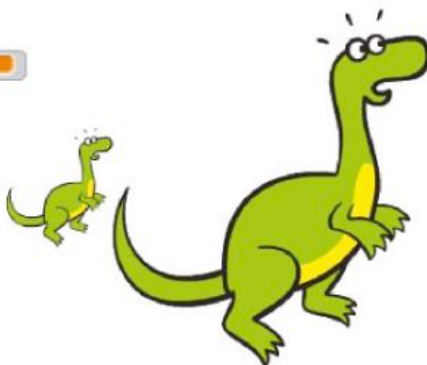
.....

.....

Exercice 4 : Je complète un programme

Compléter les zones blanches dans le programme ci-contre afin que celui-ci permette d'obtenir un programme qui affiche un lutin à une taille de 100%, et qui demande ensuite un coefficient d'agrandissement pour afficher le même lutin, sans effacer le premier, à une taille correspondante à l'agrandissement demandé.

coef 3



```

quand [drapeau] est cliqué
aller à x: -50 y: 0
mettre à 100 % de la taille initiale
demander Donner un coefficient d'agrandissement et attendre
mettre coef à [ ]
estampiller
aller à x: 100 y: 0
mettre à [ ] * [ ] % de la taille initiale
estampiller
    
```



Exercice 5 : J'ordonne un programme

Remettre dans l'ordre les instructions ci-dessous afin de réaliser un programme dans Scratch qui permette au lutin *fusée* de décompter de 10 à 0 avant le décollage.

mettre compteur à 10

ajouter à compteur -1

quand est cliqué

répéter 11 fois

dire compteur pendant 1 secondes

dire Décollage pendant 5 secondes



Instructions :

.....

.....

.....

.....

.....

.....

.....

Exercice 6 : Je complète un programme à structure imposée

Compléter le programme ci-dessous de façon à obtenir un programme qui demande un sujet, demande ensuite un verbe conjugué, puis demande un complément et, enfin, reconstitue la phrase en regroupant sujet, verbe conjugué et complément.

.....

.....

.....

.....

.....

.....

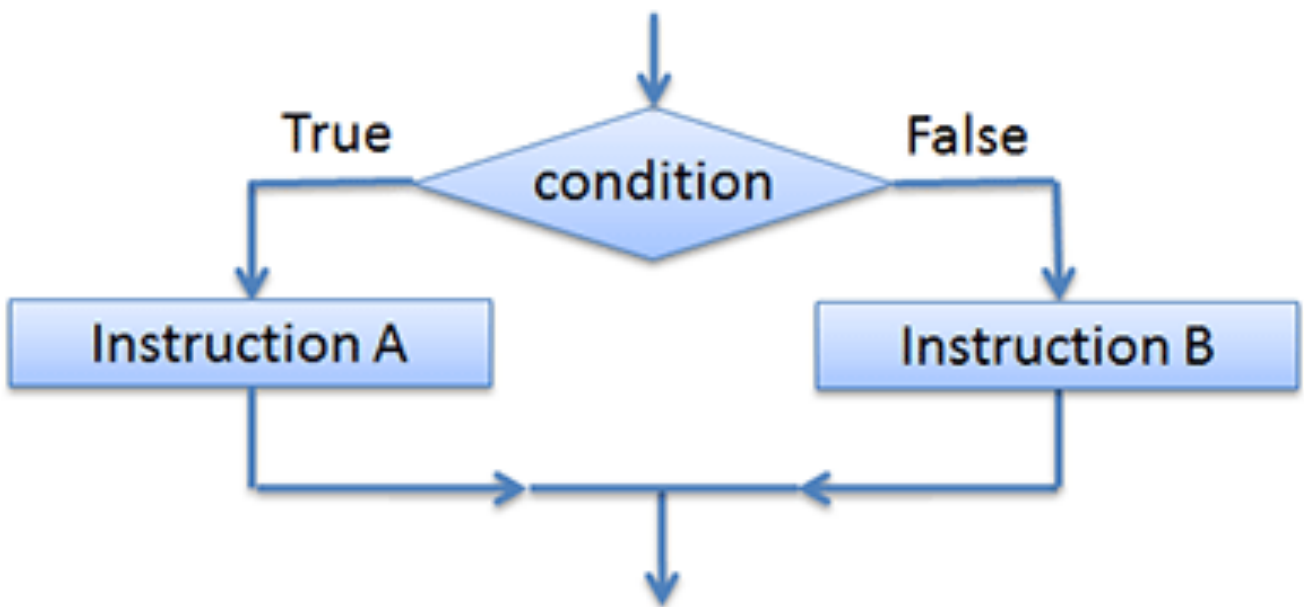
.....

.....

Séquence N°4

-

Les instructions conditionnelles





Je découvre

Une **instruction conditionnelle** est une instruction qui permet de tester une condition, puis d'effectuer une (ou plusieurs) action(s) en fonction du résultat « VRAI » ou « FAUX » de ce test. Elle s'utilise sous la forme d'un bloc d'instructions.

Exemple 1 : Un jeune conducteur se déplace en ville au volant de son véhicule :

Lorsqu'il arrive à une intersection avec un feu tricolore, il s'arrête lorsque le feu est rouge.

Si le feu est rouge
Alors le conducteur s'arrête



Exemple 2 : On organise une loterie avec des boules de différentes couleurs dans une boîte opaque.

On tire une boule au hasard sans connaître sa couleur. Si la boule est blanche, le joueur a gagné un lot.

Si la boule tirée est blanche
Alors le joueur a gagné
Sinon il a perdu



Coup de pouce

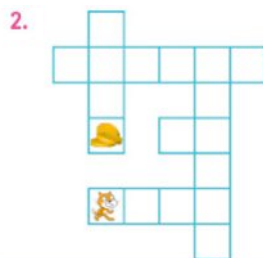
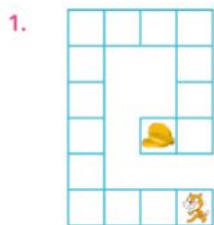
Pour tester la condition étudiée, on utilise des opérateurs de comparaison tels que « INFÉRIEUR À » ; « ÉGAL À » ; « SUPÉRIEUR À »



J'applique

Exercice 1 : Le labyrinthe

Dans chaque labyrinthe, aider Scratchy à récupérer les pièces d'or en complétant les programmes correspondant, en utilisant des instructions conditionnelles et les instructions *Avancer*, *Tourner à gauche* et *Tourner à droite* pour se déplacer à l'aide d'une boucle.



Répéter jusqu'à « pièces d'or »

Si chemin à droite Alors

Répéter jusqu'à « pièces d'or »

Si chemin à droite Alors

Exercice 2 : L'horloge digitale

Lors de certains matchs de football, en cas d'égalité à la fin du temps réglementaire, on procède à une séance de tirs au but (5 par équipe) pour désigner le vainqueur. Si les deux équipes sont encore à égalité à la fin de ces tirs au but, on continue alors en « mort-subite », c'est-à-dire jusqu'à ce qu'une équipe marque et l'autre non.

À l'aide d'instructions conditionnelles et de deux variables **buts-A** et **buts-B**, compléter l'algorithme décrivant le déroulement de cette « mort-subite ».

Instructions :

Répéter jusqu'à **buts-A** différent **buts-B**

.....

.....

.....

.....

.....

.....



Exercice 3 : Je choisis le bon programme

1. Parmi les trois programmes suivants, cocher celui qui demande un nombre et qui dit s'il est plus petit que 100 dans le logiciel Scratch.

Programme ①

```

quand [drapeau] est cliqué
demander "Donner un nombre." et attendre
mettre nombre à réponse
si nombre > 100 alors
  dire "Ce nombre est plus petit que 100." pendant 2 secondes
    
```

Programme ②

```

quand [drapeau] est cliqué
demander "Donner un nombre." et attendre
mettre nombre à réponse
si nombre < 100 alors
  dire "Ce nombre est plus petit que 100." pendant 2 secondes
    
```

Programme ③

```

quand [drapeau] est cliqué
demander "Donner un nombre." et attendre
mettre nombre à réponse
si nombre = 100 alors
  dire "Ce nombre est plus petit que 100." pendant 2 secondes
    
```

2. Expliquer brièvement pourquoi les deux autres programmes ne permettent pas de dire si le nombre testé est plus petit que 100.

.....

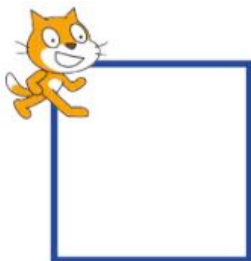
.....

.....

.....

Exercice 4 : Je complète un programme

Compléter les zones blanches dans le programme ci-contre afin que celui-ci demande une valeur, puis trace un carré ayant pour longueur de côté la valeur, mais uniquement si celle-ci est inférieure à 200.



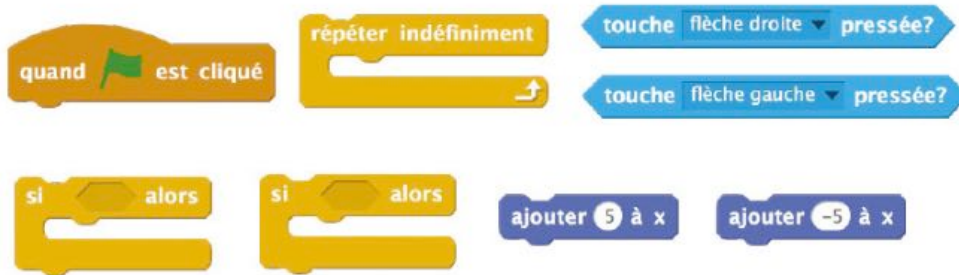
```

quand [drapeau] est cliqué
demander "Donner une longueur." et attendre
si réponse < [ ] alors
  répéter [ ] fois
    avancer de [ ]
    tourner [ ] de [ ] degrés
  
```



Exercice 5 : J'ordonne un programme

Remettre dans l'ordre les instructions ci-dessous afin de réaliser un programme dans Scratch qui permette de déplacer le lutin à droite et à gauche.



Instructions :
.....
.....
.....
.....
.....

Exercice 6 : Je complète un programme à structure imposée

Compléter le programme ci-dessous qui permet de simuler le lancer de deux dés à 6 faces, sui annonce ensuite la somme des deux dés obtenus, puis qui dit finalement « CRAPS ! » si la somme est de 7.



.....
.....
.....
.....
.....
.....
.....

Séquence N°5

-

Les blocs d'instructions





Je découvre

Un **bloc** est une **suite d'instructions** qui permet de remplacer plusieurs commandes par une seule. Dans le logiciel Scratch, l'onglet Bloc permet de définir les suites d'instructions souhaitées.

Exemple 1 : Leïla souhaite tracer un carré de 5 cm de côté.

Elle répète 4 fois le processus suivant :

- elle utilise sa règle pour tracer un segment de 5 cm ;
- elle utilise son équerre pour faire un angle droit à l'extrémité de ce segment.

Pour un carré de côté 50 pixels, on peut créer le bloc ci-contre.

Un bloc d'instructions peut être paramétré, c'est-à-dire qu'il contient au moins une instruction qui contient une information à préciser pour utiliser ce bloc.



Exemple 2 : Leïla souhaite tracer un carré de longueur donnée.

Elle répète 4 fois le processus suivant :

- avec sa règle elle trace un segment de la longueur choisie ;
- elle utilise son équerre pour faire un angle droit à l'extrémité de ce segment.

Pour tracer un carré dont le côté a une longueur précisée dans le programme, on peut créer le bloc ci-contre.

Coup de pouce

Dans ce cas, si on définit la longueur par 50, l'instruction **carré 50** donnera un carré de 50 pixels de côté.



J'applique

Exercice 1 : Le 4x100 mètres nage libre

Une épreuve de natation olympique pour laquelle l'équipe de France masculine est parmi les meilleures du monde est le 4 x 100 mètres nage libre. Cette épreuve de relais est disputée dans une piscine de 50 m de longueur par des équipes de nageurs, qui nagent chacun leur tour u aller-retour. Décrire cette course par un programme court qui utilise le bloc « **Nager son 100 mètres** » défini ci-contre.

Instructions :

Premier nageur sur le plot de départ

Nager son 100 mètres

Nageur suivant sur le plot de départ

Arrêter le chronomètre

Bloc Nager son 100 mètres :

Définir Nager son 100 mètres

Plonger dans la piscine
Nager 50 m en ligne droite
Toucher le bord de la piscine
Faire demi-tour
Nager 50 m en ligne droite
Toucher le bord de la piscine



Exercice 2 : Je choisis le bon programme

- Parmi les trois programmes suivants, cocher celui qui permet de réaliser un triangle équilatéral dont les côtés mesurent 100 pixels.

Programme ①

```

définir triangle
stylo en position d'écriture
répéter 3 fois
  avancer de 100
  tourner de 120 degrés
    
```

Programme ②

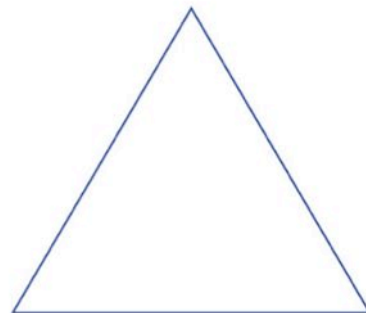
```

définir triangle
stylo en position d'écriture
répéter 3 fois
  avancer de 120
  tourner de 100 degrés
    
```

Programme ③

```

définir triangle
stylo en position d'écriture
répéter 3 fois
  avancer de 100
  tourner de 60 degrés
    
```



- Expliquer brièvement pourquoi les deux autres programmes ne permettent pas de réaliser un triangle équilatéral dont les côtés mesurent 100 pixels.

.....

.....

.....

.....

Exercice 3 : Je complète un programme

Compléter les zones blanches des instructions du programme ci-contre afin que celui-ci récite la table de 7 (de 7 x 1 à 7 x 10).

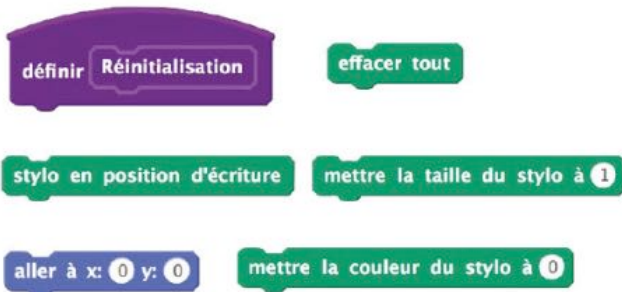
```

définir Table de 7
mettre Facteur à 0
répéter [ ] fois
  ajouter à Facteur [ ]
  dire [ ] * Facteur pendant 1 secondes
    
```



Exercice 4 : J'ordonne un programme

Remettre dans l'ordre les instructions ci-dessous afin de créer un bloc d'initialisation dans Scratch qui permette au lutin de revenir au centre, d'effacer tout et d'être en position d'écriture avec une taille 1 et une couleur rouge pour le stylo.



Instructions :

.....

.....

.....

.....

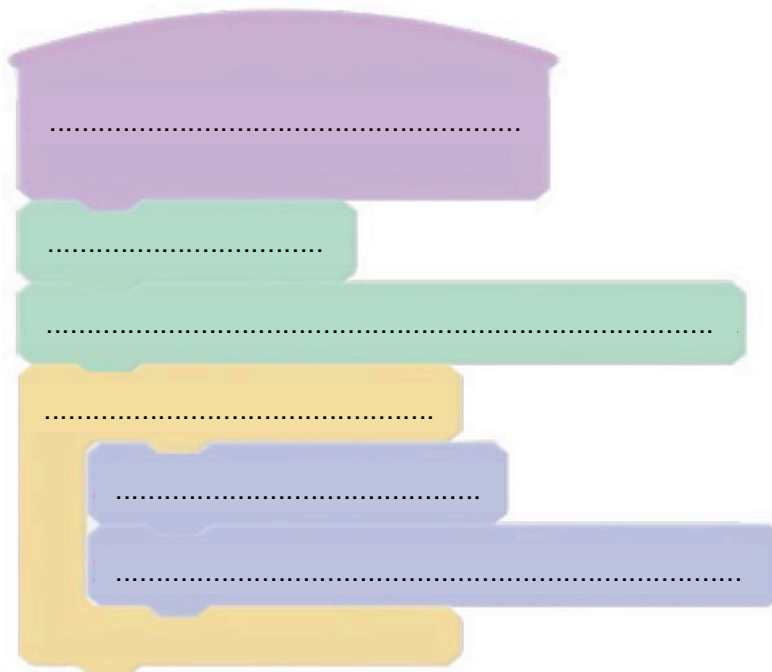
.....

.....

.....

Exercice 5 : Je complète un programme à structure imposée

Compléter la programmation du bloc ci-dessous de façon à obtenir un programme qui trace un décagone régulier (c'est-à-dire polygone à 10 côtés égaux) de 50 pixels de côté.



J'évalue mon travail



Mes compétences en algorithmique et programmation

	Insuffisant	Fragile	Satisfaisant	Confirmé
Critères de notation	Je n'ai pas fait le travail demandé. Je n'ai rien pu faire sans aide. Même avec une aide, je ne parviens pas à créer mes algorithmes.	Le travail est réalisé partiellement. Le travail réalisé comporte beaucoup d'erreurs. Je manque d'autonomie dans l'élaboration de mes algorithmes.	Le travail demandé est réalisé intégralement (avec des erreurs mineures). Le travail est réalisé dans sa majorité ET sans erreur. Je fais preuve d'autonomie.	Le travail demandé est réalisé intégralement et sans erreur. Je fais preuve d'autonomie.

Séquence 1 - Les instructions

	Insuffisant	Fragile	Satisfaisant	Confirmé
Niveau 1				

Séquence 2 - Les boucles

	Insuffisant	Fragile	Satisfaisant	Confirmé
Niveau 1				

Séquence 3 - Les variables

	Insuffisant	Fragile	Satisfaisant	Confirmé
Niveau 1				

Séquence 4 - Les instructions conditionnelles

	Insuffisant	Fragile	Satisfaisant	Confirmé
Niveau 1				

Séquence 5 - Les blocs d'instructions

	Insuffisant	Fragile	Satisfaisant	Confirmé
Niveau 1				



Mon attitude pendant les séances informatiques

	1	2	3	4
Critères de notation	Je n'ai pas eu une attitude correcte pendant la séance. Je n'ai pas fait le travail demandé. Mon livret n'est pas rempli.	J'ai eu une attitude convenable pendant la séance. Le travail demandé est fait partiellement. Mon livret est rempli partiellement.	J'ai été sérieux pendant toute la séance. Le travail demandé a été réalisé dans sa quasi-totalité et reporté intégralement dans mon livret.	J'ai été sérieux pendant toute la séance. Tout le travail demandé a été réalisé et reporté intégralement dans mon livret.

Date	Élève	Prof

Date	Élève	Prof

Date	Élève	Prof