

SCÉNARIO 2 · SI · 7^e

DÉCOUVERTE DE SCRATCH





PLAN D'ÉTUDES ROMAND

EN 22 – S'approprier les concepts de base de la science informatique...

4 ... en créant, en exécutant, en comparant et en corrigeant des programmes

Algorithmes et programmation

Création et comparaison de programmes avec des séquences, des tests conditionnels et des boucles à l'aide d'un langage de programmation visuel pour résoudre des problèmes simples

Découverte et création de sous-programmes pour améliorer un programme

Utilisation de paramètres pour modifier un programme

Liens disciplinaires

L1 21 – Compréhension de l'écrit ; L1 23 – Compréhension de l'oral ; L1 24 – Production de l'oral

MSN 22 – Nombres ; MSN 25 – Modélisation

SHS 21 – Relation Homme-Espace ; SHS 23 – Outils et méthodes de recherche



INTENTIONS PÉDAGOGIQUES

Dans ce scénario, il s'agit de faire découvrir aux élèves un logiciel de programmation visuelle : Scratch. Les élèves ont déjà pu se familiariser avec la programmation par l'entremise de Thymio ou de ScratchJr (une version simplifiée de Scratch) par exemple. Ils vont se confronter à une application plus complexe par les possibilités qu'elle propose mais qui respecte les mêmes démarches et dont l'environnement est proche. Les élèves vont ainsi apprendre à utiliser les boucles pour répéter plusieurs fois la même action, mais également qu'il existe différents types de programmation (événementielle et séquentielle).



DESCRIPTION DU SCÉNARIO

Après une manipulation de l'application et une institutionnalisation des premières découvertes autour des notions de base, les élèves vont être invités à réaliser plusieurs défis pour prendre en main l'environnement proposé.

SÉANCE	TITRE	RÉSUMÉ	MATÉRIEL	DURÉE
1	DÉCOUVRIR SCRATCH	Les élèves découvrent le logiciel Scratch à travers l'exploration de l'interface et de premiers essais de programmation.	• Fiche 1	45 minutes
2	SE DÉPLACER DANS SCRATCH !	Les élèves découvrent les spécificités du déplacement dans Scratch.	• Fiches 2, 2.1	45 minutes
3	DÉFIS !	Les élèves s'approprient Scratch en réalisant des défis successifs.	• Fiches 3, 3.1, 3.2, 4, 8, 8.1, 8.2, 8.3	45 minutes
4	CE N'EST PAS MOI QUI L'AI FAIT, MAIS JE PEUX LE MODIFIER (OPTIONNELLE)	Les élèves explorent des programmes déjà créés pour les modifier.	• Fiches 5, 6, 6.1, 7	45 minutes

Pour chaque séance en plus des fiches, il est nécessaire d'avoir :

- un affichage numérique ;
- un adaptateur VGA ou HDMI (tablette vers vidéo projecteur) ;
- une tablette avec Scratch installé (par groupes) ;
- une tablette pour projeter avec l'application Scratch installée.



CONSIDÉRATIONS SUR SCRATCH

Scratch est à la fois un langage de programmation graphique par blocs et une application qui permet de programmer avec ce langage.




Il a été développé par l'équipe de Mitchel Resnick, directeur du laboratoire Lifelong Kindergarten au MIT (Massachusetts Institute of Technology). Tandis que le logiciel est aujourd'hui traduit en 70 langues, le langage Scratch s'est hissé parmi les 20 langages de programmation les plus populaires au monde, en 2020. Ce logiciel est en open source, c'est-à-dire gratuit, modifiable et réutilisable.

Scratch est disponible sous forme d'application à installer sur tablette ou ordinateur, mais fonctionne aussi en ligne par le biais du site [78-S2-01](#).

CONSEILS PÉDAGOGIQUES

- Apprendre à programmer se fait en programmant, c'est pourquoi il est essentiel que chacun ait l'occasion de le faire. Nous conseillons de mettre les élèves par petits groupes (2-3 élèves par tablette).
- Tutorat dans la classe : si certains groupes sont bloqués, un tutorat peut être mis en place et des élèves ayant trouvés une solution à leur problème peuvent montrer à d'autres ce qu'ils ont découvert. Le tutorat doit être explicité aux élèves en amont de la séance. Cela permet de faciliter la gestion de la séance, à l'élève tuteur de mieux comprendre et au tuteuré de recevoir une aide ponctuelle.
- **Ce scénario suit une trame que l'on peut adapter à tout moment en fonction des contraintes matérielles, spatiales, de l'avancée des élèves, de leurs difficultés, etc. Rien n'interdit que l'ordre des séances soit remanié, que certaines parties prennent plus de temps ou au contraire ne soient pas menées.**

Séance 1 - Découvrir Scratch

	RÉSUMÉ	Les élèves découvrent le logiciel Scratch à travers l'exploration de l'interface et de premiers essais de programmation.
	MATÉRIEL	<ul style="list-style-type: none"> • Fiche 1 : légènder l'interface • Affichage numérique • Adaptateur VGA ou HDMI (tablette vers affichage numérique) • Tablette avec Scratch installé (par groupes) • Tablette pour projeter avec l'application Scratch installée
	DURÉE	45 minutes

Au cours de cette première séance, les élèves découvrent l'application Scratch et le langage qui lui est associé. De fortes similitudes existent avec ScratchJr, qu'ils ont normalement étudié au cycle 1 et en 5^e-6^e. Ils devraient donc pouvoir naviguer dans l'interface aisément. On peut cependant se référer à la version Junior si nécessaire.

Il est préférable de procéder à la vérification du matériel et à sa mise en place dans la classe avant de démarrer la séance (les tablettes doivent être chargées et connectées à Internet si besoin).



TEMPS 1.1

CONTEXTUALISER

EN COLLECTIF

5 minutes



« Aujourd'hui, nous allons apprendre à utiliser une application qui ressemble à ScratchJr que vous avez utilisée quand vous étiez plus jeunes. Elle s'appelle Scratch, et elle sert également à faire de la programmation. Qui peut nous rappeler ce que l'on peut faire avec ? »

Il s'agit ici d'établir un lien avec ce que les élèves ont vécu dans les classes précédentes en termes de programmation. On attend des élèves les réponses suivantes :

- écrire des histoires ;
- animer des personnages ;
- déplacer des personnages, des objets ;
- faire parler des personnages avec des bulles ou en s'enregistrant ;
- légènder une image ;
- etc.

On peut recenser au tableau les propositions des élèves.



« Avec Scratch, on va pouvoir faire toutes ces actions, et bien plus encore. C'est ce que nous allons découvrir maintenant. »



TEMPS 1.2

L'INTERFACE
DE SCRATCHEN COLLECTIF, EN
GROUPES DE 2-3 ÉLÈVES

20 minutes



« Pour ouvrir l'application Scratch sur votre tablette, vous devez cliquer sur cette icône. »

On montre à l'affichage numérique l'icône de l'application telle qu'elle est présentée sur l'appareil.



Icône de
l'application
Scratch



APPLICATION OU VERSION EN LIGNE

Scratch est disponible sous forme d'application mais il peut s'avérer difficile de l'installer sur certains appareils (non disponible sur certains magasins d'applications et dans quelques pays). Dans ce cas, on peut à tout moment utiliser la version en ligne, à partir d'un navigateur Internet (qui nécessite donc d'être connecté à Internet) à l'adresse suivante : [78-S2-03](#).

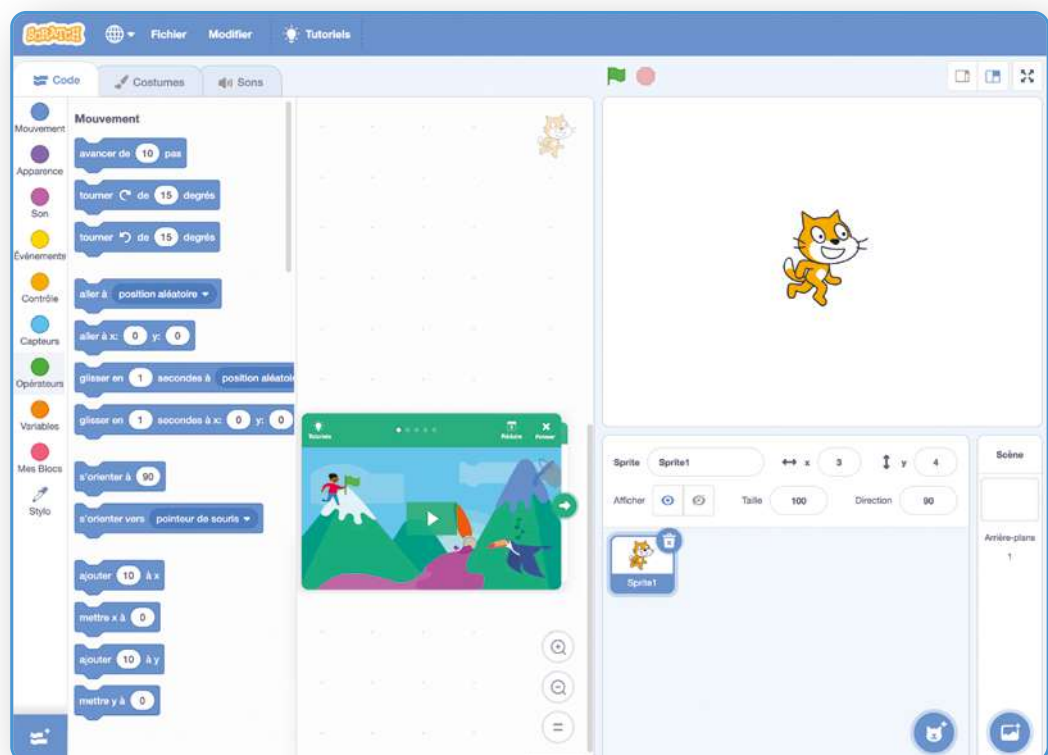
Il est possible de placer un raccourci vers cette URL sur l'écran d'accueil de la tablette.



« Lorsque l'on ouvre l'application Scratch, voici ce que l'on obtient à l'écran. »

On ouvre Scratch et on projette l'interface de l'application au tableau comme ci-dessous.

Interface de
Scratch





« Je vais vous laisser explorer librement et vous allez essayer de voir à quoi correspondent les différents éléments que vous voyez à l'écran. »

On organise des petits groupes de 2-3 élèves, en fonction du matériel disponible. On les aide pour accéder à l'application Scratch et ils peuvent démarrer l'activité. On peut aider certains groupes en leur demandant d'appuyer sur telle ou telle icône ou de déplacer certains éléments (blocs, lutin dans la scène, etc.) pour voir ce qui se passe.



« Nous allons maintenant voir ensemble ce que vous avez pu découvrir. »

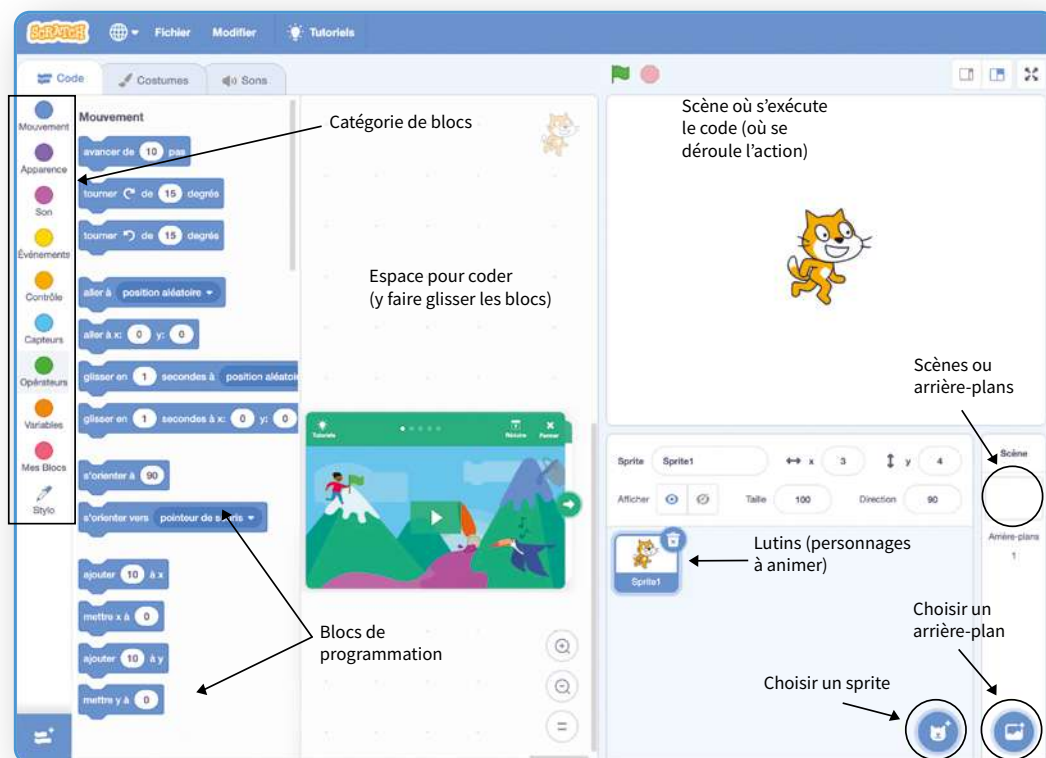
On procède après cette première exploration à une mise en commun. Pour ce faire, on peut :

- projeter l'interface de Scratch et rester sur une mise en commun orale pour demander aux élèves à quoi correspondent les différents éléments de l'interface ;
- s'appuyer sur une présentation interactive au TBI. Les élèves pourront ainsi déplacer des étiquettes et venir les placer au bon endroit, ou bien annoter directement l'image. La production collective ainsi obtenue peut être exportée au format pdf, imprimée, distribuée aux élèves pour constituer une première trace écrite ;
- conjuguer les deux propositions.

Pour mener à bien cette mise en commun, vous pouvez retrouver une présentation interactive au format « iwb », à utiliser avec les logiciels OpenBoard ou OpenSankore : [78-S2-04](#).

Il y a également à disposition la [Fiche 1](#), imprimable qui peut servir de trace écrite pour les élèves.

Présentation interactive avec interface de Scratch et étiquettes à déplacer



On veillera à ce que les élèves aient bien vu et compris comment insérer de nouveaux lutins et de nouvelles scènes en cliquant sur les boutons adéquats.

Penser à leur rappeler de sauvegarder les fichiers Scratch en cours de programmation (Fichier, Sauvegarder sur votre ordinateur).



Boutons « choisir un lutin » et « choisir un arrière-plan »



LUTIN VS SPRITE

Les objets à programmer dans Scratch s'appellent des Sprites, ce qui veut dire lutin en anglais. Nous pourrons, pour la suite de ce scénario, employer indifféremment l'un ou l'autre de ces termes.



TEMPS 1.3

CATÉGORIES
DE BLOCS

EN COLLECTIF, EN
GROUPES DE 2-3 ÉLÈVES

5 minutes



« Maintenant que nous avons vu à quoi ressemble l'application Scratch, nous allons nous intéresser aux différents blocs qui permettent de programmer. Vous avez remarqué qu'ils sont de différentes couleurs. Regardons les quatre premières catégories : mouvement, apparence, son et événements. »

On projette l'interface de Scratch et montre les quatre premières catégories avec les blocs correspondants. Il procède ensuite à une démonstration de l'utilisation de ces blocs en les faisant glisser dans la zone de programmation.



« Nous allons donc programmer le lutin avec ces catégories de blocs. Pour commencer, il nous faut un événement, c'est-à-dire que l'on décide quand le programme va démarrer. »

On fait glisser le bloc « Quand drapeau vert est cliqué ».



« Ensuite on va lui dire de faire un mouvement. »

On fait glisser le bloc « avancer de 10 pas » et on vient l'emboîter sous le premier bloc.



« On va ensuite lui dire de faire "Miaou". »

On fait glisser le bloc « jouer le son Miaou » et on vient l'emboîter sous le deuxième bloc.



« Pour finir il va changer d'apparence et on va lui demander de dire bonjour. »

On fait glisser le bloc « basculer sur le costume » et le bloc « dire Bonjour ! » et on vient les emboîter comme précédemment. C'est l'occasion de montrer l'onglet « Costumes » aux élèves et de leur faire remarquer que certains lutins peuvent changer d'apparence. On fera également observer que certains blocs sont paramétrables c'est-à-dire que l'on peut modifier leurs valeurs (« avancer de X pas » par exemple).

Premier programme



TEMPS 1.4

PETITS PROGRAMMES
DE DÉPLACEMENTS

EN GROUPES DE 2-3
ÉLÈVES

15 minutes

Suite à cette démonstration, on demande aux élèves de programmer à leur tour les déplacements de lutins.



« Vous allez essayer de déplacer un ou plusieurs lutins. Vous pouvez choisir celui que vous voulez dans la bibliothèque de lutins (bouton “ajouter un lutin”). »




On peut suggérer aux élèves de tester différents types de déplacements : de gauche à droite, de droite à gauche, de haut en bas et de bas en haut, aller et retour, en diagonal, faire le tour de la scène, etc.

Dans cette phase, on les laisse explorer librement les possibilités qui leur sont offertes. La séance suivante sera l'occasion de préciser les subtilités de Scratch dans les déplacements de personnages (répétitions, coordonnées, orientation, angles, etc.). Les élèves risquent en effet de rencontrer plusieurs problèmes :

- comment déplacer un lutin de droite à gauche (apparition de nombres négatifs) ?
- comment déplacer un lutin verticalement ?
- comment orienter un lutin différemment pour qu'il regarde dans une autre direction ?
- comment raccourcir un programme ?

Les obstacles que rencontrent les élèves peuvent être relevés sans toutefois y apporter de réponses immédiates. La recherche et le processus essais/erreurs font partie de l'apprentissage.

Séance 2 - Se déplacer dans Scratch !

	RÉSUMÉ	Les élèves découvrent les spécificités du déplacement dans Scratch.
	MATÉRIEL	<ul style="list-style-type: none"> • Fiche 2 : proposition de trace écrite • Fiche 2.1 : proposition de trace écrite (suite) • Affichage numérique • Adaptateur VGA ou HDMI (tablette vers affichage numérique) • Tablette avec Scratch installé (par groupes) • Tablette pour projeter avec l'application Scratch installée
	DURÉE	45 minutes

Il est préférable de procéder à la vérification du matériel et à sa mise en place dans la classe avant de démarrer la séance (les tablettes doivent être chargées et connectées à Internet si besoin).

Cette séance fera l'objet d'une trace écrite (voir [Fiches 2 et 2.1](#)) qui sera agrémentée au fur et à mesure des activités proposées.



TEMPS 2.1

PRÉSENTATION

EN COLLECTIF

5 minutes



« La dernière fois, nous avons vu une application de programmation. Qui peut me rappeler comment elle s'appelle et ce qu'elle permet de faire ? »

Cette question introductive permet de faire le lien avec la séance précédente et de faire un rappel du connu.

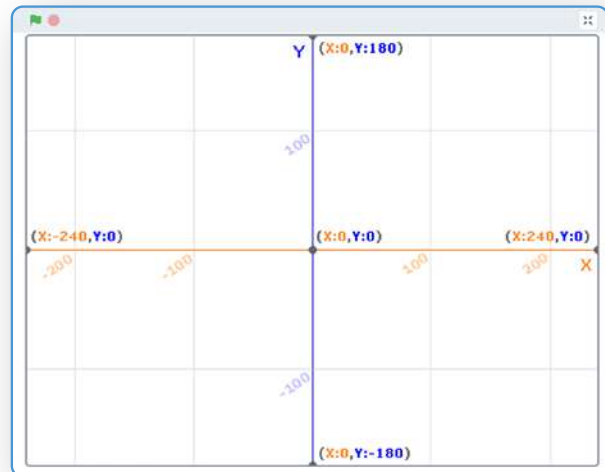


« Aujourd'hui, nous allons utiliser Scratch pour déplacer un ou des lutins. C'est ce que vous avez essayé de faire à la fin de la dernière séance et vous avez pu vous confronter à quelques difficultés que nous allons essayer de lever aujourd'hui. »

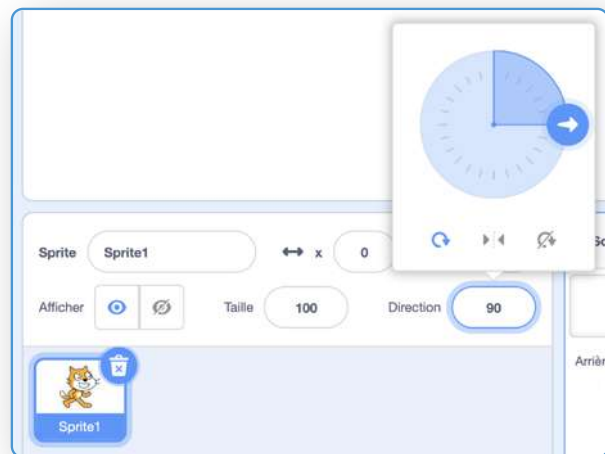
DÉPLACEMENTS DANS SCRATCH

Voici quelques informations pour comprendre l'environnement de Scratch.

1. La scène au sein de laquelle les lutins évoluent est divisée selon un système d'axes, horizontal (x) et vertical (y). On peut d'ailleurs avoir cette représentation de la scène en choisissant l'arrière-plan Xy-grid dans la bibliothèque d'arrière-plans.
2. Un lutin placé sur cette scène a une direction par défaut de 90° ce qui veut dire que le bloc « avancer de 10 pas » fera déplacer le lutin de 10 pas de gauche à droite.
3. Il n'y a pas de bloc « reculer de 10 pas », ce qui veut dire que si l'on veut que le lutin se déplace de droite à gauche, il faut attribuer une valeur négative à la variable « X pas ». Si l'on veut garder une valeur positive, il faudra changer la direction du lutin à -90° mais cela le mettra à l'envers. On peut changer la direction du lutin à -90° sans que le lutin soit mis à l'envers si on passe en mode « Gauche/Droite » (icône du milieu avec les deux triangles) au lieu de « Tout autour » (rotation).
4. Une autre solution pour déplacer un lutin est d'utiliser le bloc « ajouter une valeur à x » pour un déplacement horizontal, ou le bloc « ajouter une valeur à y » pour un déplacement vertical.



Représentation des axes x et y dans Scratch



Orientation du lutin



Valeur négative pour faire avancer le lutin de droite à gauche en gardant une direction à 90°

Toutes ces considérations étant difficiles à appréhender, nous choisissons de simplifier le discours à tenir aux élèves :



« Pour déplacer un lutin de gauche à droite, on utilise le bloc « avancer de » ou « ajouter une valeur à x » en précisant un nombre positif (« en + »). Pour le déplacer de droite à gauche, on utilise le même bloc « avancer de » ou « ajouter une valeur à x » en précisant un nombre négatif (« en - »).

Pour déplacer un lutin de bas en haut, on utilise seulement le bloc « ajouter une valeur à y » en précisant un nombre positif (« en + »). Pour le déplacer de haut en bas, on utilise ce même bloc en précisant un nombre négatif (« en - »). »



TEMPS 2.2

DÉPLACER UN LUTIN
DE DROITE À GAUCHEEN GROUPES DE 2-3
ÉLÈVES, EN COLLECTIF

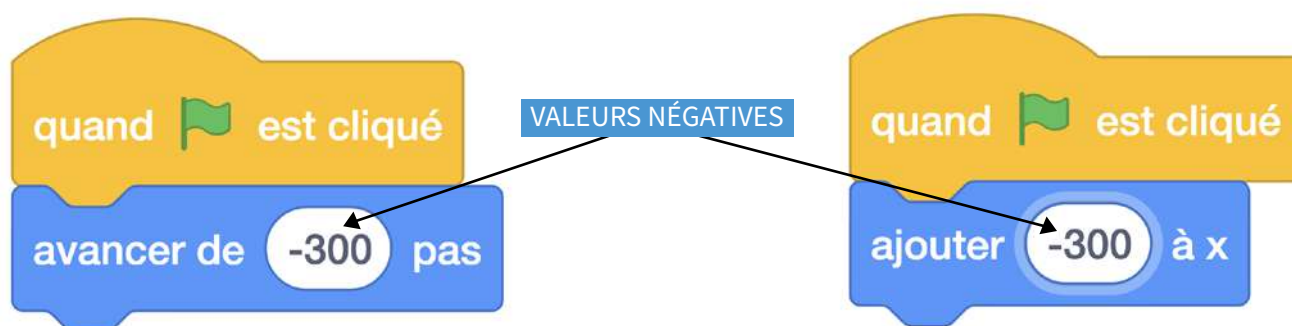
10 minutes



« Vous devez placer un lutin sur la droite de la scène et il doit se déplacer vers la gauche. »

Durant cette phase, on laisse les élèves explorer les différents blocs et tester les solutions qu'ils proposent. Lorsqu'un groupe trouve la marche à suivre, on les invite à la partager au reste de la classe.

Si au bout d'une dizaine de minutes, aucun groupe n'a trouvé de solution, on leur présente les deux solutions possibles à l'aide de l'affichage numérique. On effectue les deux programmes ci-dessous devant les élèves.



Programmes pour déplacer un lutin de droite à gauche

La trace écrite proposée aux élèves [Fiche 2](#) reprend les programmes pour les déplacements de gauche à droite et de droite à gauche.



COMPRENDRE LES DÉPLACEMENTS DANS SCRATCH À L'AIDE DES MATHÉMATIQUES

Le déplacement de la droite vers la gauche (ou du haut vers le bas) est l'occasion d'aborder les nombres relatifs (sans préciser le terme) qui sont nécessaires dans l'utilisation de Scratch. On peut éventuellement se référer à l'analogie avec les températures et le thermomètre (où les nombres négatifs se trouvent en bas). Ils peuvent donc être évoqués comme étant nécessaires pour réaliser le mouvement inverse à celui observé lorsqu'on utilise un nombre positif.



TEMPS 2.3

DÉPLACER UN LUTIN
DE DROITE À GAUCHEEN GROUPES DE 2-3
ÉLÈVES, EN COLLECTIF

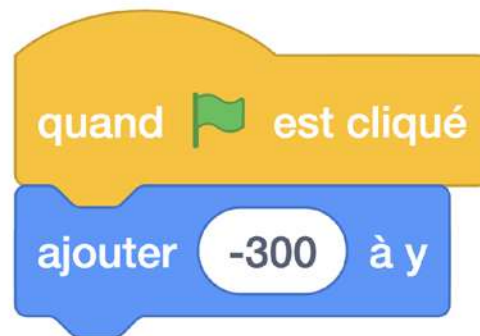
10 minutes



« Vous devez déplacer un lutin verticalement, de haut en bas et de bas en haut. »

Durant cette phase, on laisse les élèves explorer les différents blocs et tester les solutions qu'ils proposent. Lorsqu'un groupe trouve la marche à suivre, on les invite à la partager au reste de la classe.

Si au bout d'une dizaine de minutes, aucun groupe n'a trouvé de solution, on leur présente une solution possible à l'aide de l'affichage numérique. On effectue les deux programmes ci-dessous devant les élèves.



Programmes pour déplacer un lutin verticalement

La trace écrite proposée aux élèves (Fiche 2) reprend ces programmes pour les déplacements verticaux.



TEMPS 2.4

ORIENTER UN
LUTIN DANS UNE
AUTRE DIRECTIONEN GROUPES DE 2-3
ÉLÈVES, EN COLLECTIF

10 minutes



« Vous devez orienter le lutin pour qu'il regarde dans la direction inverse.
Pour cela, vous devez chercher dans l'onglet "Costumes". »

Durant cette phase, on laisse les élèves explorer l'onglet en question et tester les solutions qu'ils proposent. Lorsqu'un groupe trouve la marche à suivre, on les invite à la partager au reste de la classe.

Si au bout de quelques minutes, aucun groupe n'a trouvé de solution, on leur présente la démarche à l'aide de l'affichage numérique. On effectue la modification « orienter différemment un lutin » de la Fiche 2.1.

La trace écrite proposée aux élèves (Fiche 2.1) reprend cette démarche.



TEMPS 2.5

RACCOURCIR UN PROGRAMME

EN GROUPES DE 2-3
ÉLÈVES, EN COLLECTIF

10 minutes



« Je vous montre un programme sur ma tablette, à votre avis, que va faire le lutin ? »

On affiche le programme ci-dessous et les élèves expriment leurs hypothèses. Elles sont ensuite vérifiées en exécutant le programme devant eux.



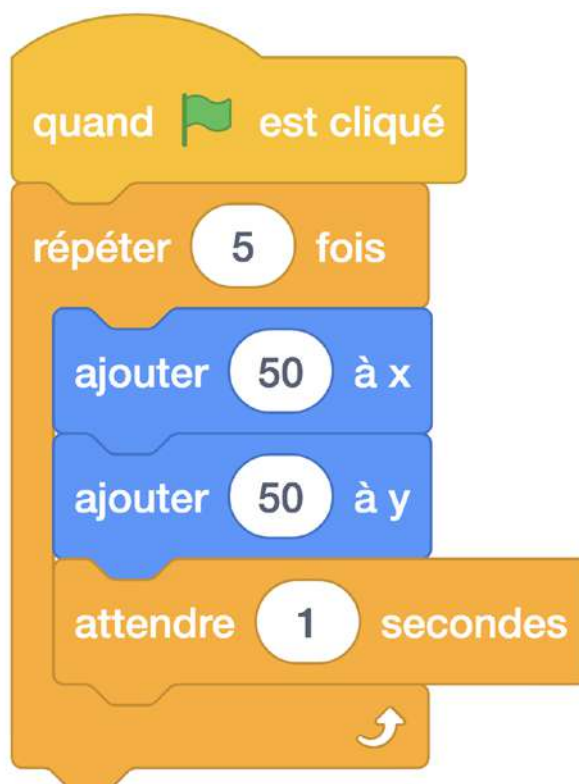
Programme pour déplacer un lutin en diagonale



« Vous allez reproduire ce programme sur vos tablettes, en petits groupes. Vous avez sûrement remarqué qu'il y a beaucoup d'instructions. Vous devez trouver une solution pour que le lutin fasse la même chose avec un programme plus court. »

Durant cette phase, on laisse les élèves explorer et tester les solutions qu'ils proposent. Lorsqu'un groupe trouve la marche à suivre, on les invite à la partager au reste de la classe.




Si au bout de 7-8 minutes, aucun groupe n'a trouvé de solution, on leur présente le bloc « répéter X fois » à l'aide de l'affichage numérique. On effectue le programme ci-dessous devant les élèves.



Programme plus court pour déplacer un lutin en diagonale

La trace écrite proposée aux élèves (Fiche 2.1) reprend cette démarche et on précise aux élèves que l'on appelle cette répétition une **boucle**.

Séance 3 - Défis !

	RÉSUMÉ	Les élèves s'approprient Scratch en réalisant des défis successifs.
	MATÉRIEL	<ul style="list-style-type: none"> • Fiche 3 : démarche pour dessiner avec un crayon • Fiche 3.1 : démarche pour dessiner avec un crayon (suite) • Fiche 3.2 : démarche pour dessiner avec un crayon (fin) • Fiche 4 : exemples de programmes pour les défis dessins • Fiches 8-8.3 : bilan • Affichage numérique • Adaptateur VGA ou HDMI (tablette vers affichage numérique) • Tablette avec Scratch installé (par groupes) • Tablette pour projeter avec l'application Scratch installée
	DURÉE	45 minutes



TEMPS 3.1

POUR DESSINER

EN COLLECTIF, EN
GROUPES DE 2-3 ÉLÈVES

5 minutes



« Vous allez réaliser aujourd'hui des programmes pour dessiner. Pour cela nous allons ajouter une catégorie de blocs. »

Avec l'affichage numérique, on montre aux élèves comment ajouter l'extension « stylo » pour ajouter de nouveaux blocs, comment ajouter un lutin « crayon » et comment modifier son costume pour qu'il dessine avec la pointe.

Vous trouverez en annexe les [Fiches 3, 3.1 et 3.2](#) expliquant la démarche, elles peuvent être distribuées aux élèves pour qu'ils puissent la mettre en œuvre sur leurs tablettes, après démonstration à l'ensemble de la classe.



TEMPS 3.2

DÉFIS DESSINS

EN GROUPES DE 2-3
ÉLÈVES

20 minutes



« Je vais maintenant vous proposer des défis. Sur la scène, vous allez essayer de tracer un certain nombre d'objets. Voici les défis à réaliser. »

Les défis sont affichés au tableau. Chaque groupe avance à son rythme :

- choisir le crayon comme lutin et dessiner un carré ;
- choisir le crayon comme lutin et dessiner un rectangle ;
- choisir le crayon comme lutin et dessiner un escalier ;
- choisir le crayon comme lutin et dessiner un cercle.

Pour aider les élèves, on s'appuie sur les exemples de programmes [78-S2-15](#) proposés sur la

[Fiche 4](#).



COMPRENDRE LES DÉPLACEMENTS DANS SCRATCH À L'AIDE DES MATHÉMATIQUES

Ces tracés peuvent venir prendre appui sur les propriétés géométriques des figures. Par exemple, la propriété du carré « quatre côtés de longueur égale » peut être convoquée pour utiliser une boucle et répéter quatre fois la même instruction. On aura également besoin de savoir que les rectangles, et donc les carrés, possèdent quatre angles droits pour orienter le stylo à 90°. De même, la réalisation du cercle peut faire appel à la mesure d'angle dont on se sert pour faire tourner le crayon d'un degré 360 fois.

Enfin, le placement du crayon, pour qu'il puisse avoir la place de dessiner, sollicite les notions mathématiques de coordonnées (x et y, abscisses et ordonnées). On y retrouve l'utilisation des nombres relatifs.



PROGRAMMER LE DÉPLACEMENT DU LUTIN

Quand on déplace un lutin avec le bloc « avancer de X pas » ou « ajouter une valeur à x », le lutin se déplace instantanément. Pour rendre le déplacement du lutin visible, on peut utiliser le bloc « glisser en X secondes à ... » ou bien marquer des pauses entre les différentes étapes de dessin avec le bloc de contrôle « attendre X secondes ». Cette dernière solution est utilisée dans les exemples de programmes de la Fiche 4.



Blocs « glisser en X secondes à ... » et « attendre X secondes »



TEMPS 3.3

DÉFIS
ARRIÈRE-PLANS

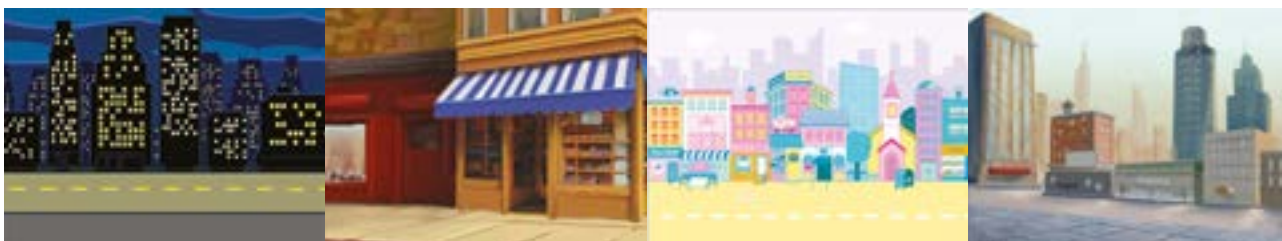
EN GROUPES DE 2-3
ÉLÈVES

10 minutes



« Vous allez maintenant écrire un programme pour que l'on puisse naviguer entre plusieurs scènes. On doit pouvoir, avec une commande, passer d'une scène à une autre. »

Dans un premier temps, les élèves vont devoir choisir différents arrière-plans parmi ceux proposés par l'application. Ci-dessous, un exemple de scènes de paysages urbains.



Exemples d'arrière-plans urbains

Le choix des types de scène et de leur nombre est laissé à la libre appréciation des élèves. On peut également demander aux élèves de cacher ou de montrer le personnage.



Afficher-masquer

Les élèves peuvent arriver à l'écriture d'un programme de type :



Arrière-plan suivant

On peut également ajouter une contrainte en demandant aux élèves d'écrire les instructions pour que l'on puisse aller directement au paysage de son choix. Sur des tablettes sans clavier physique, les événements déclenchant sont limités (on ne peut pas utiliser le bloc « quand la touche X est pressée »). On peut alors s'appuyer sur des lutins présents dans la scène qui déclenchent le passage d'un arrière-plan à un autre quand on clique dessus.



Pour changer d'arrière-plan

En fonction du niveau des élèves, il est possible de complexifier ce défi, en voulant, par exemple, depuis une page d'accueil, afficher les différents arrière-plans possibles. On peut aussi leur demander d'inclure une possibilité de revenir à la page d'accueil depuis n'importe quelle scène choisie.



TEMPS 3.4

BILAN




EN INDIVIDUEL

10 minutes

Les [Fiches 8, 8.1, 8.2 et 8.3](#) sont un bilan que vous pouvez réaliser à la suite de cette séance ou plus tard. Cette évaluation fait le point sur les principales caractéristiques de Scratch qui ont été abordées lors des trois premières séances de ce scénario.

Les fiches correspondantes peuvent s'ajouter aux fiches de travail des séances précédentes et constituent les traces écrites de l'ensemble de la séquence.

Séance 4 - Ce n'est pas moi qui l'ai fait, mais je peux le modifier (optionnelle)

	RÉSUMÉ	Les élèves explorent des programmes déjà créés pour les modifier.
	MATÉRIEL	<ul style="list-style-type: none"> • Fiche 5 : exemple de programme « Smiley » • Fiche 6 : instructions pour le programme « Étoiles de mer » • Fiche 6.1 : instructions pour le programme « Étoiles de mer » (suite) • Fiche 7 : démarche pour concevoir le programme « Changer de route » • Affichage numérique • Adaptateur VGA ou HDMI (tablette vers affichage numérique) • Tablette avec Scratch installé (par groupes) • Tablette pour projeter avec l'application Scratch installée
	DURÉE	45 minutes

Il est préférable de procéder à la vérification du matériel et à son déploiement dans la classe avant de démarrer la séance (les tablettes doivent être chargées et connectées à Internet si besoin).

Cette séance est optionnelle, c'est-à-dire que les séances précédentes sont suffisantes pour s'approprier les bases de Scratch. Si vous souhaitez aller plus loin, vous pouvez mener cette quatrième séance.

Les élèves doivent modifier des programmes déjà existants. **Il est donc nécessaire que l'enseignante ou l'enseignant les « fabrique », les enregistre et les mette à disposition des élèves. En annexe (Fiches 5, 6, 6.1, 7), se trouvent des exemples de programmes à implémenter en Scratch.**



TEMPS 4.1

SMILEY

EN GROUPES DE 2-3 ÉLÈVES

15 minutes

Le programme est sur la [Fiche 5](#). Il est divisé en quatre suites d'instructions permettant de repérer plus facilement la partie à modifier (celle de la bouche). Les élèves l'exécutent pour en observer le résultat et ils doivent ensuite chercher les éléments à modifier.



« Vous allez ouvrir, sur votre tablette, le programme qui s'appelle "Smiley", et vous allez exécuter ce programme. Que se passe-t-il ? Regardez comment est conçu ce programme. »

On demande aux élèves de décrire le résultat de l'exécution du programme, à savoir le dessin d'un smiley triste. On les questionne également sur la manière dont le programme est écrit afin d'en comprendre la logique.



« Vous allez devoir modifier ce programme pour obtenir un smiley qui sourit. »

quand je reçois Bouche ▾

aller à x: -49 y: -49

s'orienter à 30

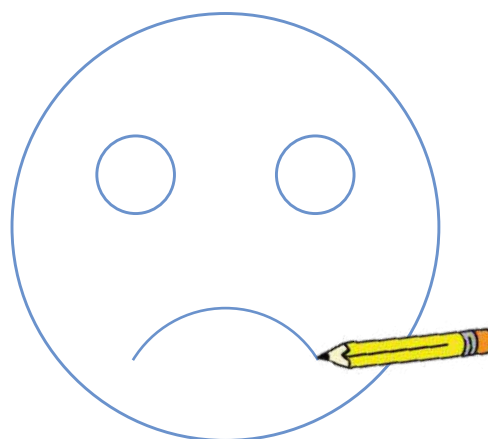
stylo en position d'écriture

répéter 120 fois

avancer de 1 pas

tourner de 1 degrés

relever le stylo



Smiley triste

Dans la suite d'instructions qui concerne la bouche, les éléments à modifier sont l'orientation du crayon, pour que celui-ci ne parte pas vers le haut mais vers le bas (« s'orienter à 150° ») ainsi que le sens de rotation qui va être antihoraire (« tourner de -1° »). Les élèves peuvent arriver à ce résultat en explorant, en testant plusieurs valeurs et en observant à chaque fois les modifications apportées dans le dessin. Pour éviter de redessiner tout le smiley à chaque fois, indiquer aux élèves qu'ils peuvent exécuter seulement la suite d'instructions pour la bouche en cliquant sur son titre « quand je reçois (bouche) ».



TEMPS 4.2

RAMASSER LES
ÉTOILES DE MEREN GROUPES DE 2-3
ÉLÈVES, EN COLLECTIF

15 minutes

Les [Fiches 6 et 6.1](#) représentent les différentes instructions pour concevoir le programme à mettre à disposition des élèves. Sa création par l'enseignante ou l'enseignant nécessite de choisir un arrière-plan (plage), d'insérer des personnages et d'en modifier les costumes.



« De la même manière que pour le smiley, vous allez ouvrir le programme qui s'appelle « Étoiles de mer ». Vous regardez l'écriture du programme, vous l'exécutez et vous observez ce qui se passe. ».

Les élèves peuvent verbaliser le fait que le garçon ramasse les étoiles de mer présentes sur la plage mais en oublie une.



« Vous devez donc modifier le programme pour faire en sorte que toutes les étoiles de mer soient ramassées. »

Dans la suite d'instructions qui concerne le personnage, il manque un déplacement vers la dernière étoile de mer et l'envoi d'un message (message 5) pour que celle-ci disparaisse. On peut ajouter facilement cette suite d'instructions en dupliquant la dernière partie et en modifiant les coordonnées de la dernière étoile ainsi que le numéro du message.



Le garçon sur la plage

Programme à modifier

NOTION DE SOUS-PROGRAMME (AVANCÉ) :

On peut profiter de ce temps pour expliquer aux élèves la notion de sous-programme dans Scratch, ceux-ci

étant créés par la commande

Créer un bloc

présente dans la catégorie [Mes Blocs](#).

Par exemple, si l'on crée un nouveau bloc

sousprog

et que lors de sa création on dit quelles sont ses

instructions en les mettant sous un chapeau

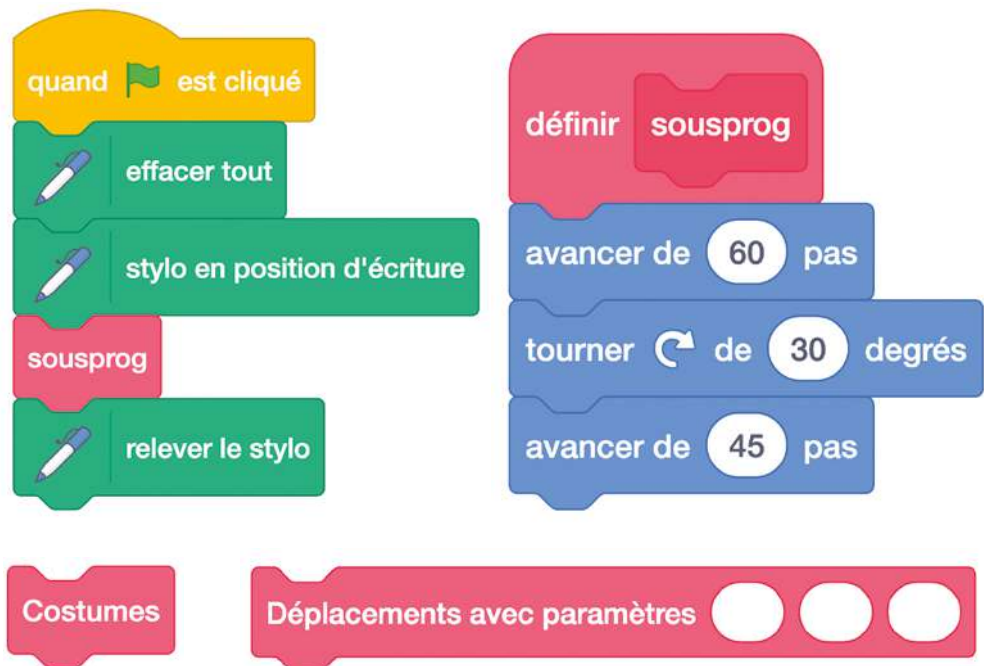
définir sousprog

alors on peut ensuite utiliser

sousprog

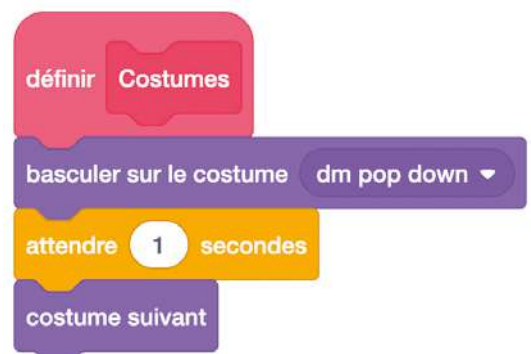
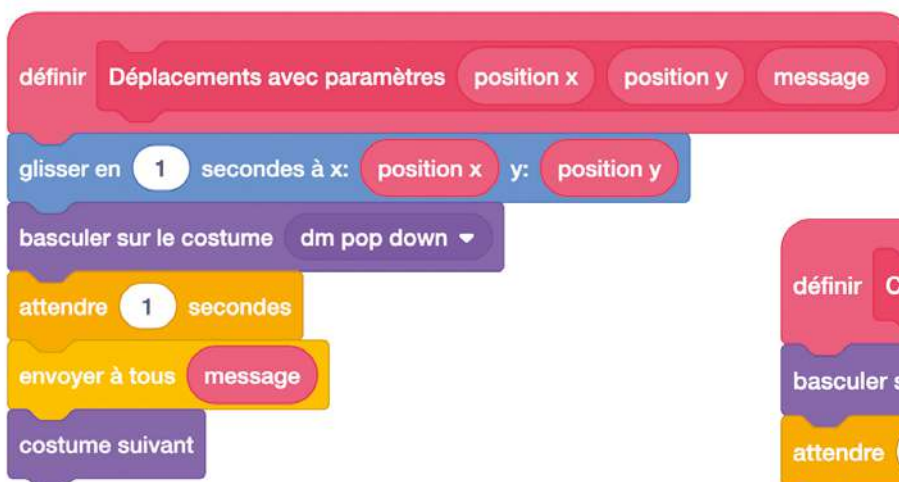
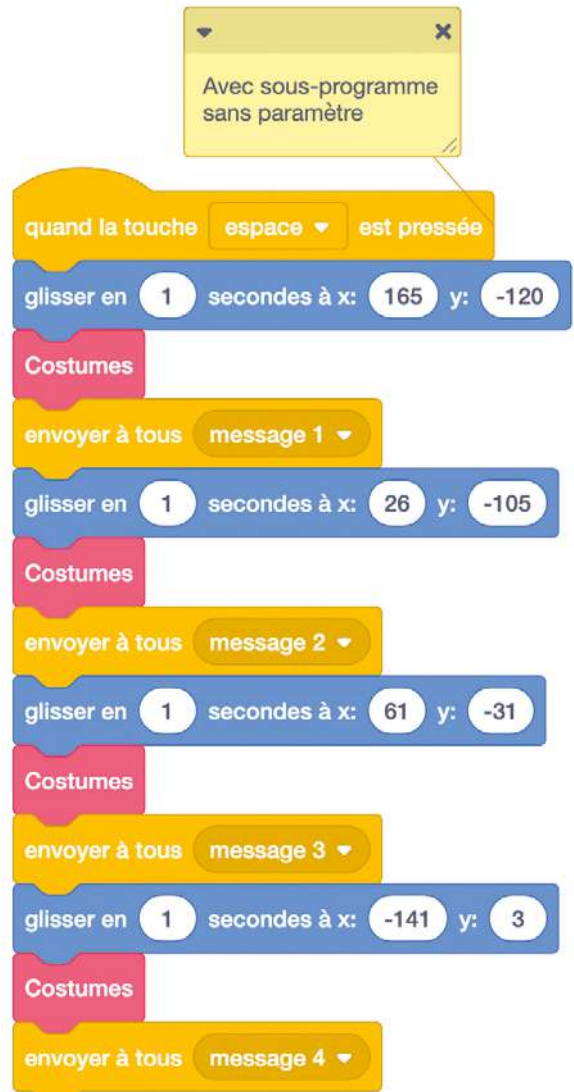
où l'on veut.

Exemple d'utilisation



En reprenant le programme « Étoiles de mer », nous pouvons créer deux blocs :

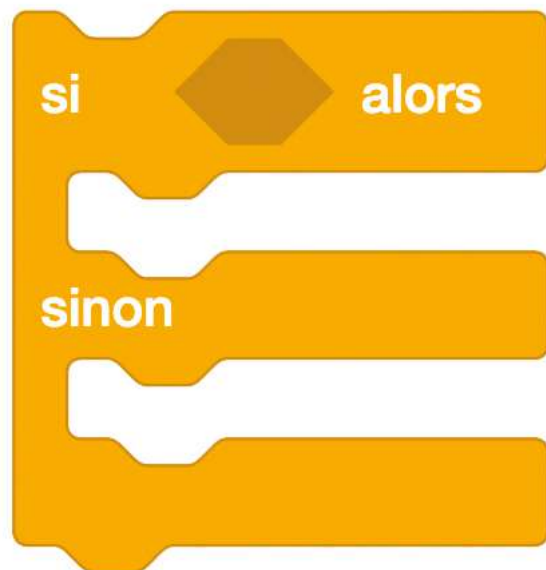
- l'un sans paramètre ;
- l'autre qui transmet des paramètres.



REMARQUE SUR LE TYPE DE PROGRAMMATION UTILISÉ (AVANCÉ) :

- programmation séquentielle : déroulement des instructions du programme les unes à la suite des autres ;
- programmation événementielle : à l'inverse de la programmation séquentielle, des événements extérieurs viennent déterminer ou modifier la séquence d'instructions en cours.

En programmation séquentielle, on peut cependant traiter des conditions pour effectuer des actions avec les blocs suivants :



Blocs d'instructions conditionnelles



TEMPS 4.3

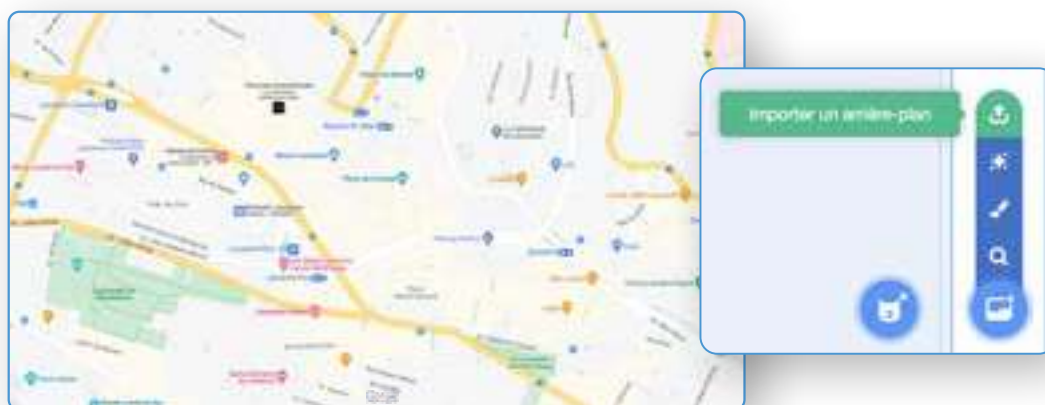
CHANGER DE ROUTE

EN GROUPES DE 2-3
ÉLÈVES

15 minutes

Voici les instructions pour concevoir le programme à présenter aux élèves.

Trouver sur Internet une image de plan de ville (ou photo de type Google Maps) et l'enregistrer sur l'ordinateur ou la tablette. Ci-dessous un exemple avec un plan simplifié de la ville de Lausanne. Si l'on veut que cette image remplisse toute la scène, il faut que cette dernière ait une définition de 480 x 360 pixels, ou un multiple de ces dimensions (960 X 720, 1 440 X 1080, etc.). On importe ensuite cette image dans Scratch grâce au bouton « Importer un arrière-plan ».



Projeter la [Fiche 7](#).



« Vous ouvrez maintenant le programme qui s'appelle "Plan Lausanne". Soyez attentifs à l'écriture du programme, vous l'exécutez et vous observez ce qui se passe. »

Les élèves peuvent verbaliser le fait que Scratch-chat (le lutin par défaut) se déplace dans Lausanne, part de la place Saint-François pour aller à la place de la Riponne, en passant par la place de la Palud.



« Vous allez modifier le programme pour qu'il passe cette fois-ci par la rue du Grand-Pont, la place Bel Air et la place Saint-Laurent. »



CHOIX DES SUPPORTS

Les exemples ci-dessus sont donnés à titre indicatif. L'enseignant peut choisir une image ou un plan adapté au contexte local et/ou modifier les trajets présentés.

RESSOURCES ET LIENS :

Informatique créative [78-S2-11](#) est la traduction en français du guide Creative Computing (link is external) de l'Université de Harvard. La traduction de ce document a été réalisée par Inria (link is external), Institut de recherche Français de sciences et technologies numériques.

Toutes les commandes Scratch : [78-S2-12](#)

Ressource CANOPÉ, « De la logique au code avec Scratch Junior » : [78-S2-13](#)

La palette des blocs Scratch : [78-S2-14](#)

Légender l'interface



The image shows the Scratch interface with several key areas highlighted and labeled:

- Scène où s'exécute le code (où se déroule l'action):** Points to the main stage area where the cat sprite is currently positioned.
- Scènes ou arrière-plans:** Points to the 'Scène' and 'Arrière-plans' tabs in the top right corner.
- Choisir un sprite:** Points to the 'Sprites' button in the top right corner.
- Choisir un arrière-plan:** Points to the 'Backgrounds' button in the top right corner.
- Lutins (personnages à animer):** Points to the 'Sprites' panel on the right, which shows the 'Sprite1' selected.
- Code:** Points to the 'Code' tab in the bottom left corner.
- Catégorie de blocs:** Points to the 'Mouvement' category in the bottom left corner.
- Espace pour coder (y faire glisser les blocs):** Points to the central workspace area where code blocks are dropped.
- Blocs de programmation:** Points to the 'Mouvement' block palette on the left side of the workspace.

The 'Mouvement' block palette includes the following blocks:

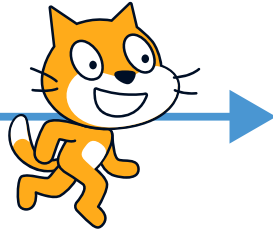
- avancer de 10 pas
- tourner ⤴ de 15 degrés
- tourner ⤵ de 15 degrés
- aller à position aléatoire
- aller à x: 0 y: 0
- glisser en 1 secondes à position aléatoire
- glisser en 1 secondes à x: 0 y: 0
- s'orienter à 90
- s'orienter vers pointeur de souris
- ajouter 10 à x
- mettre x à 0
- ajouter 10 à y
- mettre y à 0

Proposition de trace écrite



DÉPLACEMENTS HORIZONTAUX

POUR ALLER DE GAUCHE À DROITE



quand est cliqué

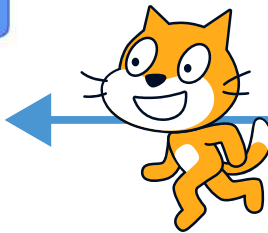
avancer de **300** pas

NOMBRES POSITIFS

quand est cliqué

ajouter **300** à x

POUR ALLER DE DROITE À GAUCHE



quand est cliqué

avancer de **-300** pas

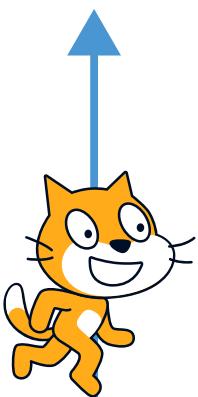
NOMBRES NÉGATIFS

quand est cliqué

ajouter **-300** à x

DÉPLACEMENTS VERTICAUX

nombres positifs



POUR ALLER DE
BAS EN HAUT

quand est cliqué

ajouter **300** à y

nombres négatifs

quand est cliqué

ajouter **-300** à y

POUR ALLER DE
HAUT EN BAS

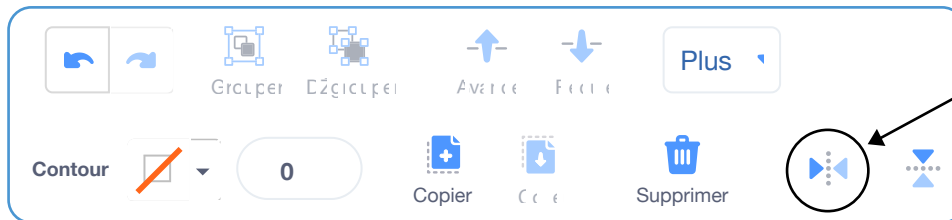
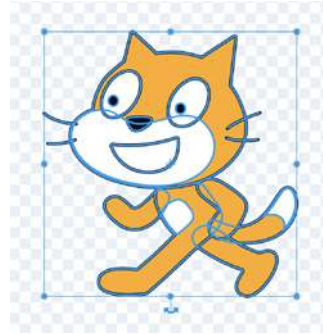
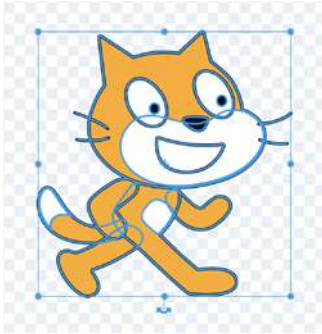


Proposition de trace écrite (suite)



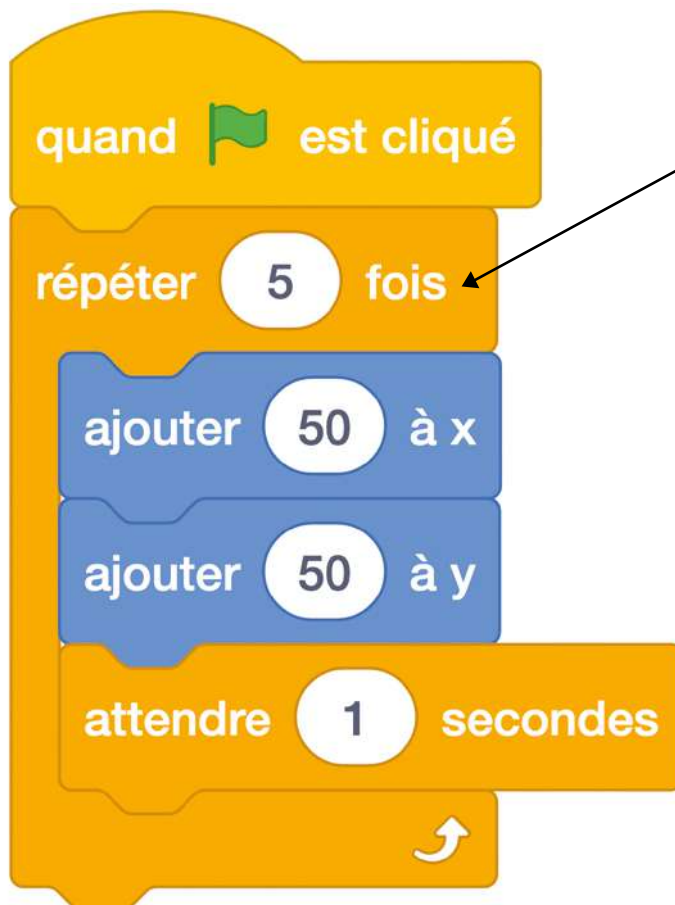
ORIENTER DIFFÉREMMENT UN LUTIN

Sélectionner l'ensemble du lutin



Cliquer sur
« Retourner
horizontalement ».

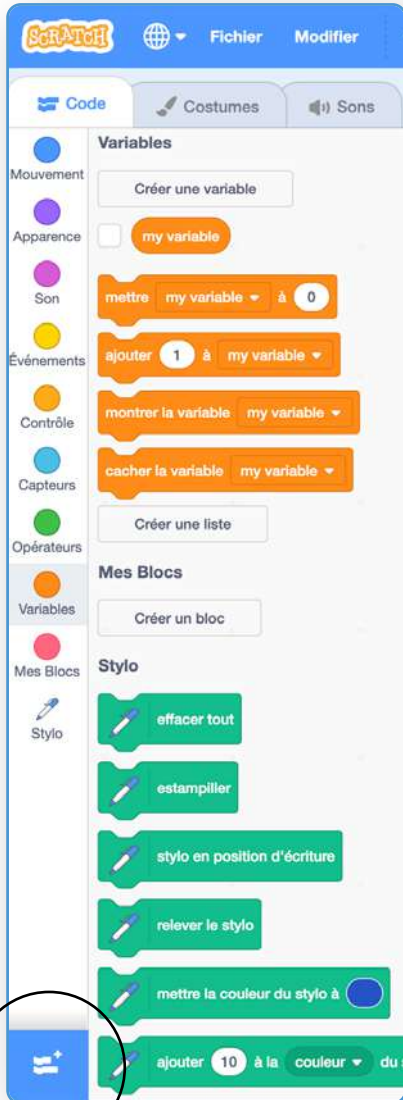
RACCOURCIR UN PROGRAMME



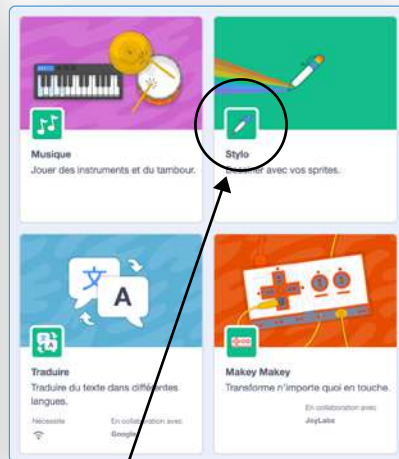
Si des instructions se répètent, je peux utiliser le bloc « répéter X fois » pour raccourcir le programme. On appelle cela une **boucle**.

Démarche pour dessiner avec un crayon

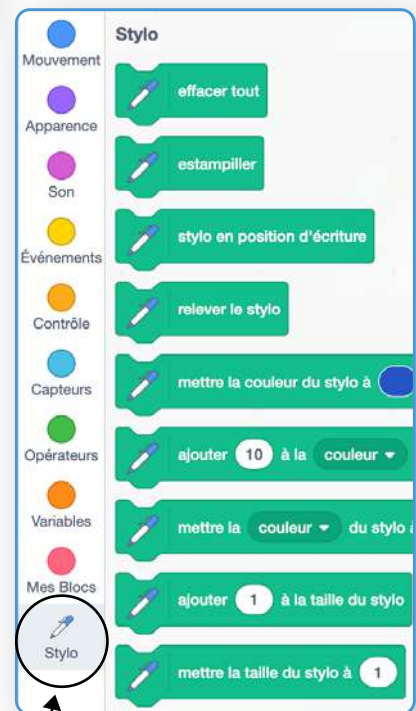
POUR AFFICHER LES BLOCS « STYLO »



1 Cliquer ici pour ajouter des catégories de blocs.



2 Puis choisir la catégorie « Style ».

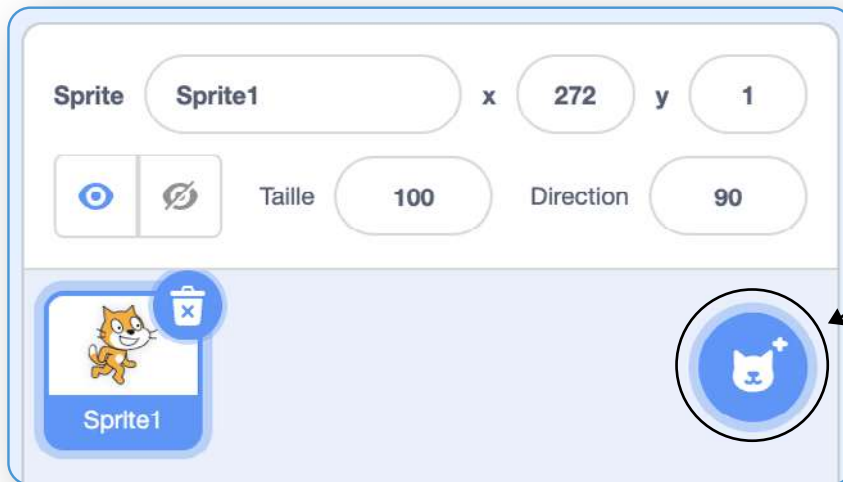


3 De nouveaux blocs apparaissent.

Démarche pour dessiner avec un crayon (suite)



POUR CHOISIR LE LUTIN « CRAYON » (PENCIL)



1 Pour afficher le stylo, cliquer sur « choisir un sprite ».

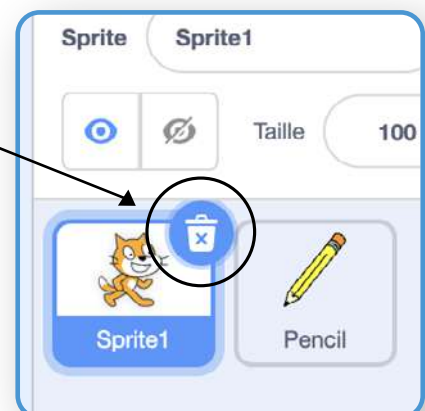
Faire défiler tous les lutins pour trouver le stylo, ou taper « Pencil » dans la barre de recherche.

2



3

Pour supprimer le chat, cliquer dessus puis sur la corbeille qui apparaît en haut à droite du lutin.





Démarche pour dessiner avec un crayon (fin)

POUR CENTRER LA MINE DU CRAYON

- 1 Cliquer sur l'onglet « Costumes ».
- 2 Cliquer sur l'outil « Flèche ».
- 3 Avec l'outil, sélectionner tout le crayon en faisant glisser la flèche.
- 4 On peut ensuite déplacer le crayon pour que la mine corresponde au point de visée.
- 5 On peut aussi diminuer sa taille avec les poignées du cadre.

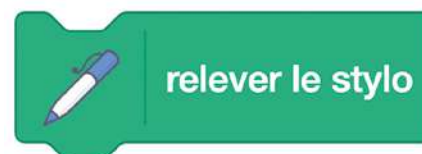
POUR ÉCRIRE OU DESSINER

Pour écrire ou dessiner, il faut placer le bloc « Stylo en position d'écriture ».

Chaque déplacement du lutin laissera ainsi une trace sur la scène.

Si on veut déplacer le lutin sans qu'il laisse de trace, il faut placer l'instruction « relever le stylo ».

Si l'on veut effacer toutes les traces, il faut placer le bloc « effacer tout ».





Exemples de programmes pour les défis dessins

DÉFI DESSINS 4:
EXEMPLE POUR
UN CERCLE

```

    quand est cliqué
    effacer tout
    aller à x: 0 y: 120
    stylo en position d'écriture
    répéter 360 fois
    avancer de 1 pas
    tourner de 1 degrés
    relever le stylo
  
```

DÉFI DESSINS 3:
EXEMPLE POUR
UN ESCALIER

```

    quand est cliqué
    effacer tout
    aller à x: -211 y: -127
    stylo en position d'écriture
    répéter 10 fois
    avancer de 30 pas
    tourner de 90 degrés
    attendre 0.2 secondes
    avancer de 30 pas
    tourner de 90 degrés
    attendre 0.2 secondes
    relever le stylo
  
```

DÉFI DESSINS 2:
EXEMPLE POUR
UN RECTANGLE

```

    quand est cliqué
    effacer tout
    aller à x: -185 y: -1
    stylo en position d'écriture
    répéter 2 fois
    avancer de 100 pas
    tourner de 90 degrés
    attendre 1 secondes
    avancer de 50 pas
    tourner de 90 degrés
    attendre 1 secondes
    relever le stylo
  
```

DÉFI DESSINS 1:
EXEMPLE POUR
UN CARRÉ

```

    quand est cliqué
    effacer tout
    aller à x: -185 y: -1
    stylo en position d'écriture
    répéter 4 fois
    avancer de 100 pas
    tourner de 90 degrés
    attendre 1 secondes
    relever le stylo
  
```

Exemple de programme « Smiley »



LA SÉPARATION EN QUATRE SUITES D'INSTRUCTIONS PERMET DE REPÉRER PLUS FACILEMENT LA PARTIE À MODIFIER (CELLE DE LA BOUCHE).

The image shows a Scratch script for a character named 'Smiley'. The script is organized into four distinct routines, each triggered by a different event. Each routine follows a similar sequence of actions: moving to specific coordinates, orienting the character, drawing a line with the pen tool, repeating a series of steps (moving forward and turning), and finally lifting the pen. The routines are: 1) 'quand cliqué' (when clicked), 2) 'Oeil gauche' (left eye), 3) 'Oeil droit' (right eye), and 4) 'Bouche' (mouth). The 'Bouche' routine is highlighted in yellow, indicating it is the focus of the document's explanation.

```

    quand cliqué
    effacer tout
    relever le stylo
    aller à x: 0 y: 120
    s'orienter à 90
    stylo en position d'écriture
    répéter 360 fois
    avancer de 2 pas
    tourner de 1 degrés
    relever le stylo
    envoyer à tous Oeil gauche

    Oeil gauche
    quand je reçois Oeil gauche
    aller à x: -46 y: 57
    s'orienter à 90
    stylo en position d'écriture
    répéter 360 fois
    avancer de 0.3 pas
    tourner de 1 degrés
    relever le stylo
    envoyer à tous Oeil droit

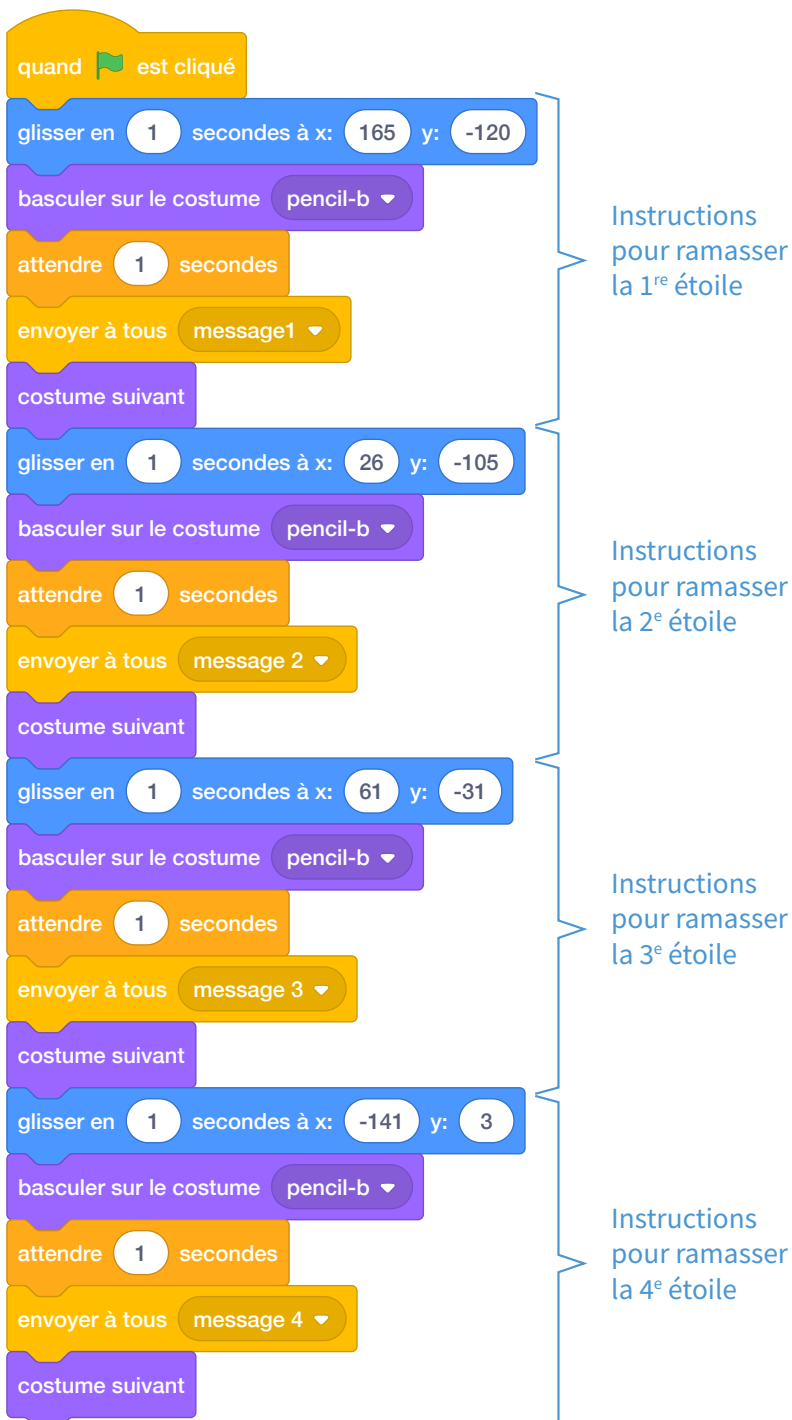
    Oeil droit
    quand je reçois Oeil droit
    aller à x: 44 y: 57
    s'orienter à 90
    stylo en position d'écriture
    répéter 360 fois
    avancer de 0.3 pas
    tourner de 1 degrés
    relever le stylo
    envoyer à tous Bouche

    Bouche
    quand je reçois Bouche
    aller à x: -49 y: -49
    s'orienter à 30
    stylo en position d'écriture
    répéter 120 fois
    avancer de 1 pas
    tourner de 1 degrés
    relever le stylo
  
```


Instructions pour le programme « Étoiles de mer »



- Choisir l'arrière-plan « Beach Malibu ».
- Choisir le lutin « D-Money-D » et le disposer sur la plage.
- Ne garder que les costumes 1 et 7, supprimer les autres.
- Réduire la taille du personnage.
- Choisir le lutin Starfish 5 fois et les disposer sur la plage.
- Réduire la taille des étoiles de mer.



Instructions pour replacer les lutins à leurs positions initiales.

Les coordonnées indiquées dans les instructions doivent correspondre aux coordonnées des étoiles de mer et de la position initiale du personnage.

Dans Scratch, contrairement à ScratchJr, il faut réinitialiser la position et l'état (affiché ou non) des lutins.

Instructions pour le programme « Étoiles de mer » (suite)



quand je reçois message1 ▼
cacher

quand je reçois message2 ▼
cacher

quand je reçois message3 ▼
cacher

quand je reçois message6 ▼
montrer

quand je reçois message6 ▼
montrer

quand je reçois message6 ▼
montrer

Instructions pour la 1^{re} étoile

Instructions pour la 2^e étoile

Instructions pour la 3^e étoile

quand je reçois message4 ▼
cacher

quand je reçois message5 ▼
cacher

quand je reçois message6 ▼
montrer

quand je reçois message6 ▼
montrer

Instructions pour la 4^e étoile

Instructions pour la 5^e étoile

Démarche pour concevoir le programme « Changer de route »

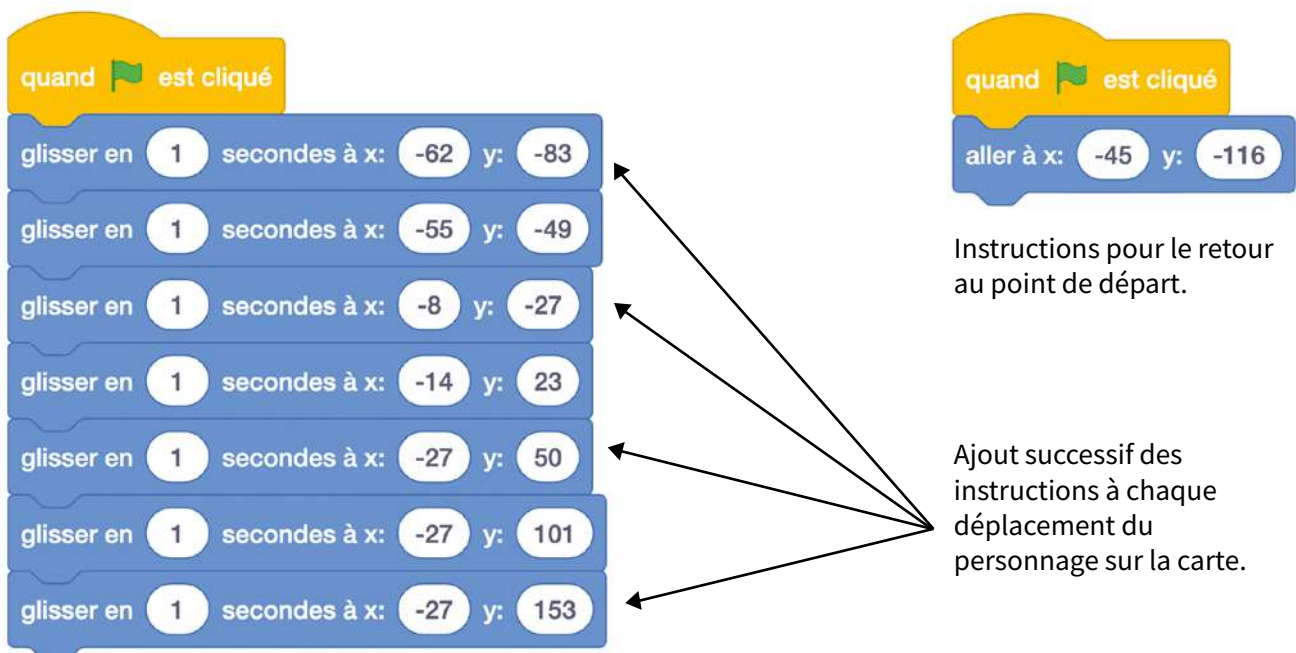


Quand on place le personnage sur le plan, les blocs « Mouvement » précisant les coordonnées du lutin prennent automatiquement les valeurs de la position (exemple ci-dessous).

aller à x: -45 y: -116

glisser en 1 secondes à x: -45 y: -116

Il suffit donc de déplacer le personnage à plusieurs reprises sur le chemin et à chaque fois d'ajouter un bloc pour coder son trajet. Il faut également prévoir une commande pour faire retourner le personnage à son point de départ.



Bilan



Dans les textes ci-dessous, place les mots suivants :

ÉVÉNEMENTS

STYLO

PROGRAMMATION

LUTIN

MOUVEMENT

LANGAGE

BLOCS



SCRATCH

PRÉSENTATION

Scratch est une application et un langage de .

Scratch permet de manipuler du code sous forme de en couleurs.

Le chat de Scratch, nommé Scratch-Cat est la mascotte du site mais aussi le
par défaut qu'on trouve dans chaque projet au moment de sa création.

DIFFÉRENTES BRIQUES DE CODAGE

Le Scratch est constitué de multiples briques (ou blocs)
permettant d'exécuter une action précise.

Il existe neuf catégories de blocs différentes (classées par couleurs) :

BLOCS « CLASSIQUES » :

-
- Apparence
- Son
-
- Contrôle
- Capteurs
- Opérateurs
- Variables
- Mes blocs

Des catégories de blocs peuvent aussi être ajoutées, comme la catégorie « »
qui permet de dessiner.

Bilan (suite)



Regarde le programme ci-dessous.



De combien de pas au total le personnage va avancer ?

Quel bloc appartient à la catégorie « Événements » ?

Quel bloc appartient à la catégorie « Mouvement » ?

Que faut-il modifier pour que le personnage se déplace de 50 pas ?



Dans les textes ci-dessous, place les mots suivants :

ÉVÉNEMENTS

STYLO

PROGRAMMATION

LUTIN

MOUVEMENT

LANGAGE

BLOCS



SCRATCH

PRÉSENTATION

Scratch est une application et un langage de **PROGRAMMATION**.

Scratch permet de manipuler du code sous forme de **BLOCS** en couleurs.

Le chat de Scratch, nommé Scratch-Cat est la mascotte du site mais aussi le **LUTIN** par défaut qu'on trouve dans chaque projet au moment de sa création.

DIFFÉRENTES BRIQUES DE CODAGE

Le **LANGAGE** Scratch est constitué de multiples briques (ou blocs) permettant d'exécuter une action précise.

Il existe neuf catégories de blocs différentes (classées par couleurs) :

BLOCS « CLASSIQUES » :

- **ÉVÉNEMENTS**
- Apparence
- Son
- **MOUVEMENT**
- Contrôle
- Capteurs
- Opérateurs
- Variables
- Mes blocs

Des catégories de blocs peuvent aussi être ajoutées, comme la catégorie « **STYLO** » qui permet de dessiner.



Regarde le programme ci-dessous.



De combien de pas au total le personnage va avancer ?

Il va avancer 10 fois de 10 pas, soit 100 pas au total.

Quel bloc appartient à la catégorie « Événements » ?

Le bloc « quand la touche espace est pressée ».

Quel bloc appartient à la catégorie « Mouvement » ?

Le bloc « avancer de 10 pas ».

Que faut-il modifier pour que le personnage se déplace de 50 pas ?

Il faut modifier le nombre de répétitions : « répéter 5 fois ». Ou il faut modifier le nombre de pas : « avancer de 5 pas ».

