

P5 – P6

10-12



Dossier pédagogique

Mission débris spatiaux & Scratch

Fiche enseignant

TABLE DES MATIÈRES

Vue d'ensemble	3
Résumé des activités	5
Contexte.....	6
Introduction	8
Partie 1 – Initiation Scratch.....	8
Partie 2 - Découverte des algorithmes et des logigrammes	12
Partie 3 - Création du jeu	15
Pour aller plus loin.....	22

La Scientothèque



VUE D'ENSEMBLE

Description

En écrivant un programme d'un jeu de récupération des débris spatiaux, les élèves découvrent le langage de programmation par blocs avec Scratch.

Objectif :

- Programmer à l'aide du logiciel de programmation "Scratch Desktop" un jeu consistant à éviter des débris suite à une explosion dans l'espace.
→ Pour voir le code final du programme du jeu, voici le lien : <https://scratch.mit.edu/projects/1212369710>

Objectifs d'apprentissage :

- Appréhender le déplacement horizontal et vertical d'un objet d'un point A à un point B
- Comprendre la notion de variables, évènements, boucle et condition en programmation
- Lire, verbaliser et écrire un programme, un algorithme et un logigramme

Notions abordées

Programmation, instructions, mathématiques, algorithme, logigramme. Scratch, débris, robot, jeu, bloc, boucle, conditions.

Tranche d'âge préconisée

10-12 ans

Durée

Entre 2h et 3h

Comprend l'utilisation de

Ordinateurs ou tablettes avec le logiciel Scratch Desktop.

Prévoir 1 ordinateur par jeune pour les groupes de 12 jeunes ou 1 ordinateur par binôme pour les groupes de 24 jeunes.

Références, liens utiles et licence

Référentiels disciplinaires

Dans le cadre du Pacte pour un enseignement d'excellence, de nouveaux référentiels d'enseignement ont été mis en place par la Fédération Wallonie Bruxelles et seront progressivement intégrés dans les programmes des écoles au cours des prochaines années. Au travers de cette activité proposée par la Rentrée des Sciences, il vous est possible de travailler certains attendus de ces référentiels avec vos élèves. Les savoirs, savoir-faire, compétences et attendus sont décrits dans le document « Liens avec les référentiels ».

Utilisation de la ressource

Les ressources mises à votre disposition dans le cadre de la Rentrée des Sciences ont été évaluées et adaptées par l'équipe de La Scientothèque – ESERO Belgium en partenariat avec l'initiative Sciences et Enseignement afin de répondre à la réalité de l'enseignement maternel et primaire. Si besoin, nous vous invitons à adapter cette ressource afin qu'elle corresponde au mieux aux spécificités de votre classe.

Remarque

Pour des raisons d'ergonomie de lecture, le texte de cette ressource pédagogique n'est pas rédigé en écriture inclusive mais il s'adresse néanmoins tant aux hommes qu'aux femmes, ainsi qu'aux personnes non-binaires.

Liens utiles

- Scratch mit : <https://scratch.mit.edu/>
- Fiche pédagogique pour s'initier au logiciel Scratch et à la programmation en blocs « Parcours Découverte Thymio & Scratch »
- Conférence Vidéo ESERO Belgium « L'espace, une opportunité et une responsabilité : l'enjeu des débris spatiaux » par le professeur Olivier Chaziot du Von Karman Institute (Belgique)
- Programme pour le jeu de base sur le site Scratch mit
- Programme pour le jeu avec options sur le site Scratch mit

Droits d'auteur

© ESERO Belgium 2025 – La Scientothèque



Le contenu de cette fiche pédagogique est publié sous la licence [Creative Commons \[CC-BY-NC-ND\]](#) : Attribution – Pas d'utilisation commerciale – Pas de modifications 4.0 International

RÉSUMÉ DES ACTIVITÉS

Activité	Titre	Description	Résultat	Exigences	Durée
1	Prise en main de Scratch	Les élèves vont découvrir les fonctionnalités de base du logiciel de programmation Scratch.	Comprendre les notions de blocs, de déplacement selon une abscisse et une ordonnée.	Aucune	1h00
2	Découverte des algorithmes et logigrammes	Les élèves découvrent les notions d'algorithmes et de logigrammes.	Comprendre les notions d'algorithmes et de logigrammes. Lire et écrire un algorithme et un logigramme.	Activité 1	1h
3	Jeu de base	Les élèves vont utiliser la programmation par bloc pour créer un jeu dans lequel un robot doit éviter des débris. Ils pourront personnaliser l'interface de leur jeu et programmeront les mouvements du robot ainsi que le placement des débris sur l'écran.	Comprendre les notions de déplacement selon une abscisse et une ordonnée, ainsi que les notions de variables, boucles et de conditions en programmation.	Activités 1 et 2	2h



CONTEXTE

L'espace est un formidable terrain d'exploration et de développement technologique, mais il souffre également de son exploitation. Les débris y prolifèrent. Ils résultent d'événements accidentels ou intentionnels. Il s'agit de pièces de satellites, vaisseaux spatiaux ou encore d'autres objets envoyés dans l'espace qui ne sont plus utilisés. Elles restent en orbite autour de la Terre et peuvent se déplacer à des vitesses atteignant les 29 000 km/h. Actuellement, des milliers de débris orbitent autour de la Terre. Ces débris ont déjà un impact et constituent une menace pour les activités spatiales présentes et futures.

Quels sont les enjeux liés aux débris spatiaux ?

Les débris spatiaux constituent un problème à différents niveaux. Pour des satellites, des fusées ou des vaisseaux spatiaux fonctionnels, le risque d'impact augmente avec le nombre de débris en orbite. Même le plus petit débris peut endommager les engins spatiaux puisqu'ils voyagent à grande vitesse. Cela peut mener à ce qu'on appelle le syndrome de Kessler. C'est un scénario théorique dans lequel, lorsque la densité de débris en orbite est assez grande et cause des impacts, cela mène à un effet de cascade puisque de nouveaux débris sont créés par ces impacts. En plus de constituer un risque pour les satellites et les vaisseaux spatiaux habités, la densité de débris spatiaux complexifie le travail des contrôleurs de mission sur Terre. En effet, ceux-ci sont amenés à réaliser des manœuvres complexes pour empêcher la collision de satellites ou vaisseaux spatiaux avec les débris. Finalement, il peut également arriver, bien que ce soit rare, que certains débris sortent de leur orbite pour s'écraser sur la Terre, représentant un danger pour la population et les animaux.

En plus de ces enjeux, le nombre croissant de débris présents dans l'espace soulève des questions éthiques. La pollution de la Terre n'est-elle pas déjà suffisante, et pouvons-nous empêcher de polluer d'autres endroits ? Ces matériaux polluants l'espace ne pourraient-ils pas être recyclés sur Terre, au lieu d'être considérés comme des ressources "perdues" ?

Pour l'instant, il n'existe pas encore de lois concernant les débris spatiaux. Cependant, étant donné qu'ils font obstacle aux agences et entreprises spatiales, ces dernières tendent à mettre en place des programmes de gestion des déchets et débris spatiaux.

Pour aller plus loin, [une page dédiée](#) à la thématique des débris spatiaux est disponible sur le site d'ESA Education.

Il est également possible de visionner la Conférence Vidéo ESERO Belgium [« L'espace, une opportunité et une responsabilité : l'enjeu des débris spatiaux »](#) donnée par le professeur Olivier Chaziot, chef du département Aérospatiale et Aéronautique de l'Institut von Karman de Dynamique des Fluides (Rhode-Saint-Genèse) qui étudie de façon théorique mais aussi expérimentale la façon dont les satellites sont détruits (ou non) en rentrant dans l'atmosphère terrestre.

Voici quelques liens donnant des informations complémentaires sur les débris spatiaux.

- https://www.esa.int/Space_in_Member_States/France/Point_de_situation_sur_les_debris_spatiaux
- <https://www.nasa.gov/headquarters/library/find/bibliographies/space-debris/>
- https://www.esa.int/Space_Safety/Space_Debris/About_space_debris
- https://fr.wikipedia.org/wiki/D%C3%A9bris_spatial
- <https://www.nature.com/articles/d41586-025-00797-7>
- <https://www.nhm.ac.uk/discover/what-is-space-junk-and-why-is-it-a-problem.html>
- <https://www.sciencedirect.com/science/article/abs/pii/S0376042122000501>
- <https://www.scu.edu/environmental-ethics/resources/on-the-pressing-issue-of-space-debris/>

INTRODUCTION

Cette fiche pédagogique se présente comme un guide pour l'enseignant ou l'animateur afin qu'il puisse accompagner les jeunes, étape par étape, dans la création et la programmation du jeu. D'un point de vue pédagogique, il est recommandé de présenter chaque étape que les jeunes doivent réaliser, de les laisser réfléchir puis de reprendre avec eux les différentes solutions trouvées afin d'en choisir une. Le programme Scratch présenté dans cette fiche est une des possibilités pour arriver au résultat visé, mais n'est pas la seule solution.

PARTIE 1 – INITIATION SCRATCH

Le principe du jeu que les jeunes vont devoir programmer est le suivant : des débris spatiaux sont placés à des points stratégiques de l'écran. Le joueur doit les éviter et atteindre le drapeau final. Pour ce faire, il doit déplacer son personnage contrôlé, ici un petit robot, en utilisant les touches fléchées du clavier.

Découverte du logiciel Scratch

Pour s'initier au logiciel de programmation, nous vous conseillons d'installer en amont le logiciel Scratch Desktop (troisième édition) sur les ordinateurs qu'utiliseront les jeunes au travers du lien suivant : <https://scratch.mit.edu/download>

Les jeunes débutent l'activité en prenant en main les fonctionnalités de base du logiciel. Pour ce faire, ils sont invités à se rendre au lien suivant pour créer leur identifiant : <https://scratch.mit.edu/join>

Une fois l'identifiant créé, les jeunes se rendent sur le logiciel Scratch installé sur l'ordinateur et se connectent.

Sur la page d'accueil, invitez les jeunes à découvrir l'environnement de programmation avec les différentes sections présentes : palette d'instructions, zone du programme, la scène, la zone « Sprite » ainsi que la zone « arrière-plan ».

Tout au long de cette partie dédiée à la phase de découverte, les élèves ont différents défis à relever.

Défi 1 – Découverte de l'interface

Ils commencent par découvrir l'interface en sélectionnant des blocs de leur choix répondant aux consignes données et les placent dans la zone de programme.

Voici les consignes attendues :

1) Ajouter et assembler/détacher des blocs.

→ à l'aide de la souris, les jeunes effectuent un clic gauche sur le bloc présent dans la zone d'instruction et le font glisser dans la zone de programme. Pour assembler les blocs, il suffit de placer un nouveau bloc en dessous de l'ancien.

2) Exécuter le programme [aussi en grand écran].

→ pour exécuter le programme, les jeunes cliquent sur le drapeau vert situé en haut à gauche de la scène. Pour mettre la scène en grand écran, il faut cliquer sur la case en haut à droite de la scène présentée par les 4 flèches se dirigeant vers l'extérieur.

3) Supprimer un ou des blocs.

→ deux possibilités :

- a) effectuer un clic gauche sur le bloc et appuyer sur la touche « delete » du clavier.
- b) effectuer un clic droit sur le bloc et cliquer sur « Supprimer le bloc ».

4) Dupliquer des blocs.

→ deux possibilités :

- a) effectuer un clic gauche sur le bloc et sur le clavier, appuyer sur la ctrl+C [copier] et ensuite Ctrl+V [coller].
- b) effectuer un clic droit sur le bloc et cliquer sur « Dupliquer ».

Défi 2 – Déplacement du Sprite

Dans ce défi, les jeunes sont invités à émettre des hypothèses sur les manières possibles de déplacer le Sprite dans la scène.

Pour déplacer le Sprite, il importe d'utiliser différents blocs.

Exemple pour se déplacer vers le haut, aller dans :

- 1) la rubrique « événements », ajouter le bloc [Quand « drapeau vert » est cliqué] dans la zone de programme.
- 2) la rubrique « contrôle », ajouter le bloc [si.. alors] dans la zone de programme.

3) la rubrique « **capteurs** », ajouter le bloc [si touche flèche haut pressée] dans la zone de programme.

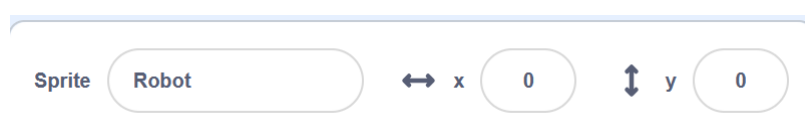
4) la rubrique « **mouvement** », ajouter le bloc [ajouter 4 à y] dans la zone de programme.

Il est également possible de déplacer le Sprite en effectuant un clic gauche sur le Sprite dans la scène et de le bouger avec la souris.

Défi 3 – Se repérer dans l'espace

Les jeunes sont amenés à placer le Sprite à des coordonnées fixées. Pour ce faire, un rappel des axes des X et des Y est à effectuer en amont.

Pour placer le Sprite au bon endroit, les jeunes peuvent indiquer les données de X et Y du tableau dans les cases en dessous de la scène :



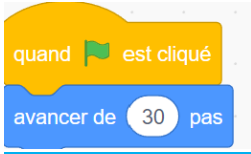
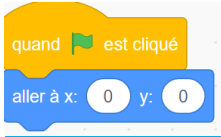
À cette étape de l'activité, les jeunes ne sont pas obligés d'utiliser des blocs à placer dans la zone de programme. Ils seront amenés à le réaliser dans la partie 2 du dossier.

Dans la seconde partie du défi 3, les jeunes devraient trouver que le Sprite se situe à X : 100 ; Y : 100.

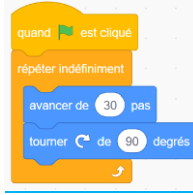
Défi 4 – Tester le programme en effectuant des consignes simples

Dans cette partie, les jeunes sont invités à suivre les étapes demandées dans la colonne de gauche du tableau en utilisant les bons blocs. Pour les aider, une relance reprenant les blocs à employer est indiquée à la page 3 du dossier élève. Toutefois, ils ne sont pas mis dans l'ordre chronologique du tableau pour laisser les jeunes effectuer des essais-erreurs.

Voici le corrigé final :

	Quel(s) bloc(s) as-tu utilisé(s) ?	Dans quelle catégorie as-tu trouvé le(s) bloc(s) ?
1) Lorsque l'on clique sur le drapeau vert , faire avancer le chat de 30 pas 		Événement Mouvement
2) Lorsque l'on clique sur le drapeau vert , mettre le chat au centre (x = 0 ; y = 0) de la scène		Événement Mouvement
3) Lorsque l'on clique sur le drapeau vert , amener le chat à dire « bonjour » 		Événement Apparence
4) Faire avancer le chat de 30 pas et lui faire dire « bonjour » lorsque l'on clique sur le drapeau vert 		Événement Apparence Mouvement
5) Lorsque l'on clique sur le drapeau vert : faire avancer le chat de 30 pas, le faire dire bonjour et le faire tourner de 90 degrés 		Événement Apparence Mouvement
6) Option A : Quand on clique sur le drapeau vert : répéter 3 fois « faire avancer le chat de 30 pas » et « le faire tourner de 90 degrés »		Événement Contrôle Mouvement

7) Option B : Quand on clique sur le drapeau vert : **répéter indéfiniment** « faire avancer le chat de 30 pas » et « le faire tourner de 90 degrés »



Événement

Contrôle

Mouvement

PARTIE 2 – DÉCOUVERTE DES ALGORITHMES ET DES LOGIGRAMMES

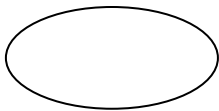


Les jeunes sont amenés à programmer, et ainsi à rédiger des instructions très précises à suivre. L'ensemble des étapes s'appelle **l'algorithme**. Il est possible de le schématiser à l'aide de **logigrammes**.

Un **algorithme** peut être comparé à une recette ou une liste d'étapes à suivre pour réaliser un élément. Il s'agit de suivre des instructions dans le bon ordre pour accomplir une mission ou réaliser une tâche.

Un **logigramme** consiste à dessiner un plan/un schéma pour montrer comment réaliser quelque chose.

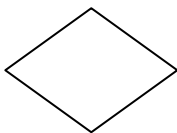
Pour expliquer les différentes étapes à suivre, nous utilisons des symboles, des formes et des flèches spéciales.



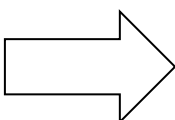
L'ovale indique le début et la fin du logigramme.



Le rectangle indique chaque étape du programme.



Le losange indique une condition, « si...alors ».



Les flèches indiquent le sens des étapes.

Défi 1 – Rédiger un algorithme en donnant des instructions à un robot

Pour s'entraîner à comprendre la notion d'algorithme, les élèves sont invités à donner des consignes simples et précises à leur robot pour qu'il arrive au drapeau.



Voici le corrigé de l'algorithme

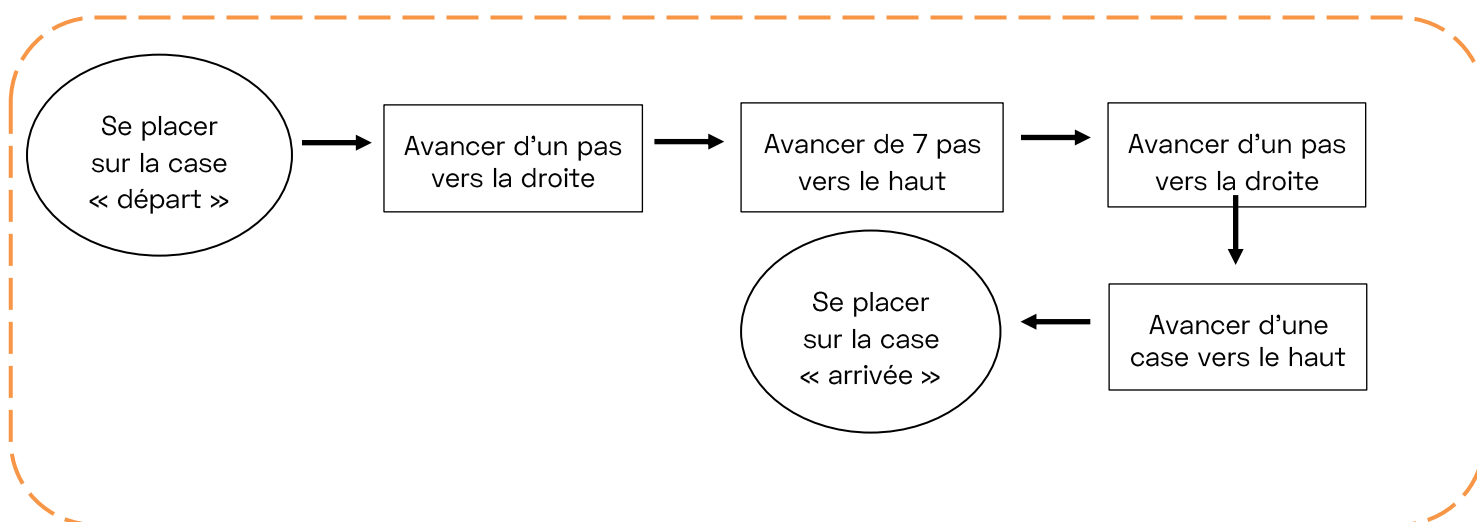
Se placer sur la case « départ ». Avancer d'une case vers la droite. Avancer de 4 cases vers le haut. Avancer d'une case vers la droite. Avancer d'une case vers le haut. Se placer sur la case « arrivée ».

NB : Il s'agit d'un exemple de réponse. Selon le niveau des jeunes, il est possible de rédiger les étapes différemment.

Défi 2 – Dessiner le logigramme de l'algorithme

Une fois la notion d'algorithme comprise, les jeunes apprennent à le schématiser à l'aide du logigramme.

Voici le corrigé du logigramme :



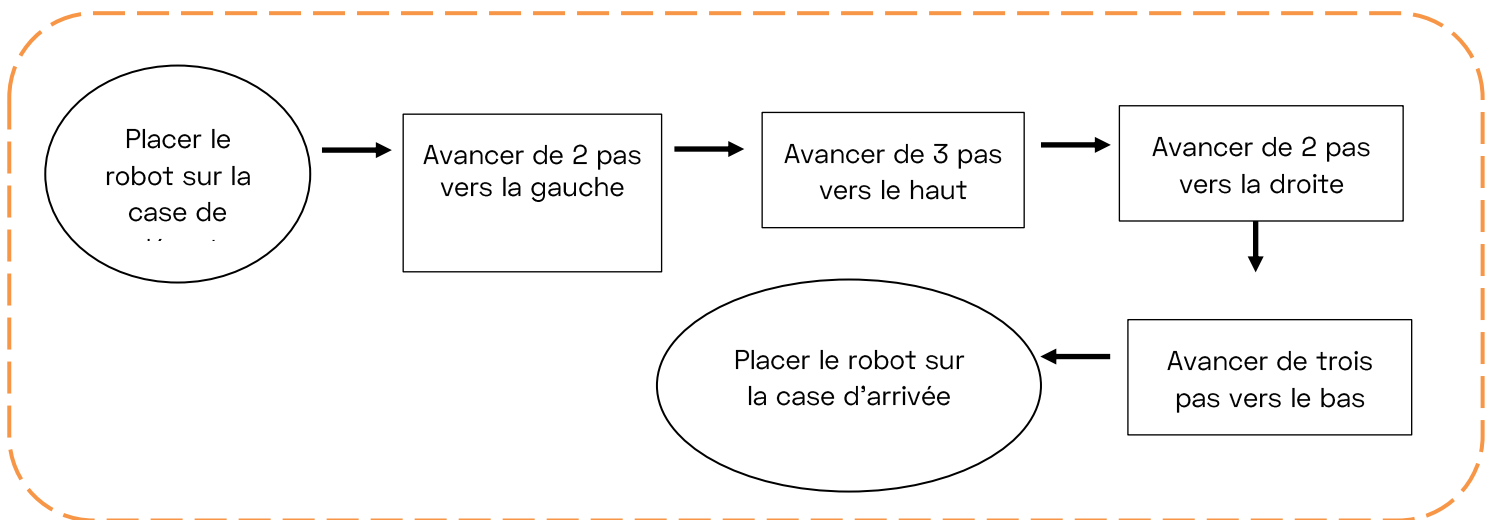
NB : Il s'agit d'un exemple de réponse. Selon le niveau des jeunes, il est possible de rédiger les étapes différemment.

Défi 3 – Dessiner un logigramme à partir d'un algorithme

Pour se familiariser avec l'apprentissage, les jeunes sont invités à s'entraîner à schématiser le logigramme d'un algorithme donné.

Placer le robot sur la case d'arrivée. Avancer de 2 pas vers la gauche. Avancer de 3 pas vers le haut. Avancer de 2 pas vers la droite. Avancer de trois pas vers le bas. Placer le robot sur la case d'arrivée.

Voici le corrigé du logigramme :



NB : Il s'agit d'un exemple de réponse. Selon le niveau des jeunes, il est possible de rédiger les étapes différemment.

PARTIE 3 – CRÉATION DU JEU

Etape 1 - Découverte des débris spatiaux.

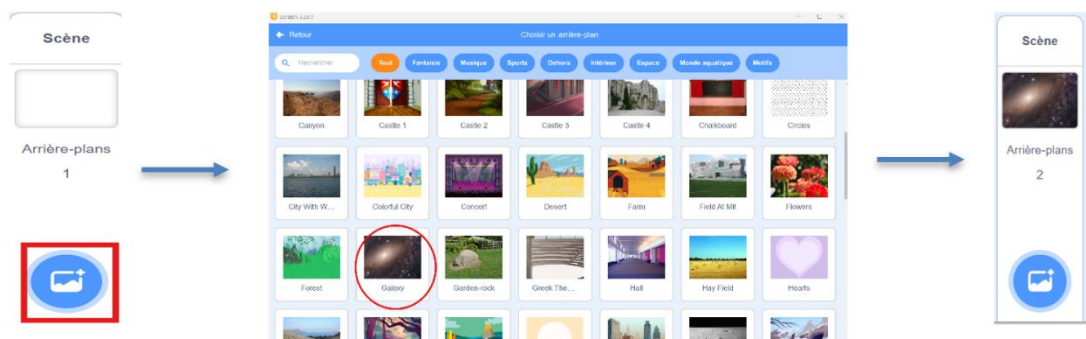
Dans cette partie, les jeunes débutent en donnant des hypothèses sur ce que sont des débris spatiaux selon eux, les manières dont ils sont créés et les impacts induits par leur présence dans l'espace.

Etape 2 – Mission de programmation

Défi 1 – Changement du décor

Étant donné que les jeunes travaillent dans un contexte spatial, ils sont invités à personnaliser l'**arrière-plan** de la scène où se déplace le Sprite.

Pour ce faire, ils se rendent dans l'espace "Scène" et cliquent sur la puce bleue représentée par une icône « image ». Ils déroulent la roulette de la souris vers le bas jusqu'à tomber sur l'arrière-plan « Galaxy » qui est disponible dans la base de données de Scratch. Une fois que l'image de la Galaxy est cliquée, le fond est changé.



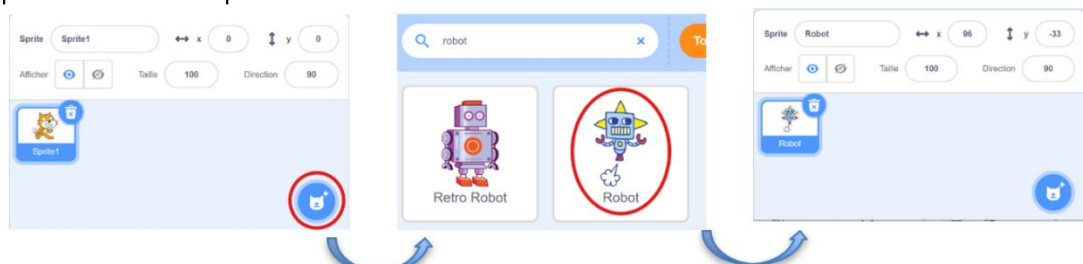
Défi 2 – Personnalisation du robot

Pour rappel, Scratch fonctionne avec des Sprites (lutins), c'est-à-dire des personnages ou objets à programmer. Il est possible de personnaliser le jeu en choisissant son Sprite. Il suffit pour cela d'utiliser les puces bleues au bas des zones Sprite et scène.

- ➔ Les jeunes cliquent sur la puce bleue représentée par un chat. Ils remplacent le Sprite par défaut (personnage de chat) par le robot (inclus dans la base de données de Scratch).



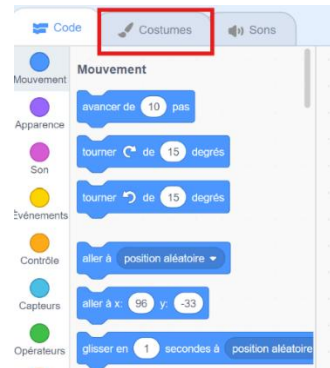
Pour faciliter la recherche, ils peuvent taper « robot » dans la barre de recherche en haut à gauche de l'écran et cliquer sur le robot à droite. Il importe de supprimer le Sprite chat pour ne laisser que le robot.



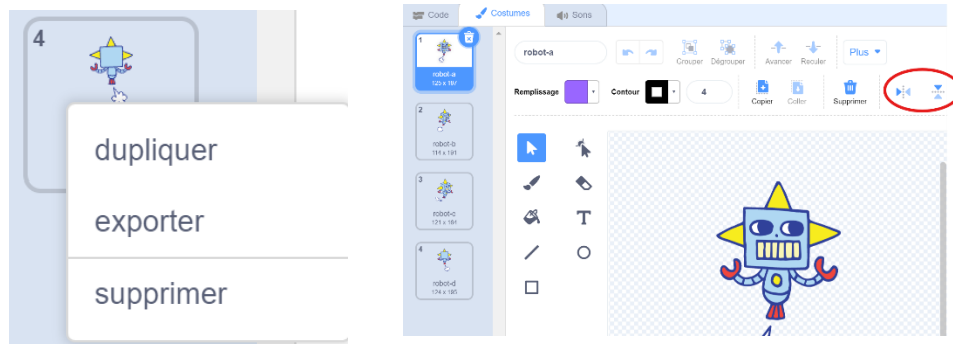
Dans le cadre de ce programme, le robot est invité à se déplacer à droite, à gauche, vers le haut et le bas. Dans ce cadre, les jeunes sont invités à **personnaliser chaque face** du robot en modifiant son costume.

Pour ce faire, ils créent différents costumes au robot dans l'onglet "Costumes".

- « robot-a » de face,
- « robot-b » de profil à gauche,
- « robot-c » de profil à droite,
- « robot-d » de dos (optionnel)



Pour créer chaque face, les jeunes effectuent un clic droit sur le robot-a et cliquent sur « dupliquer ». Pour changer la face du robot, ils cliquent sur les flèches en haut à droite de l'écran (cf. photo ci-dessous).



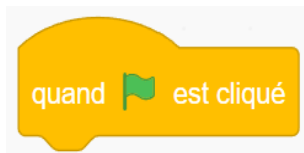
Défi 3 - Placer le Sprite au centre de la scène

Pour donner une position initiale cohérente au Sprite tout au long du programme, les jeunes sont invités à le placer à une position de base bien précise.

Nous vous suggérons de définir la position de départ dans le coin en bas à gauche de la scène. Cela correspond à $X = -210$ et $Y = -150$.

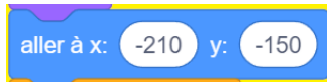
Ils se rendent dans la palette d'instructions et saisissent les blocs suivants :





le code.

Le bloc «quand drapeau vert est cliqué» permet de lancer automatiquement le programme dès que le joueur appuie sur le drapeau vert. Il sert de point de départ à l'exécution du code et assure que toutes les instructions commencent en même temps, de manière organisée. Il importe donc de le placer en premier dans

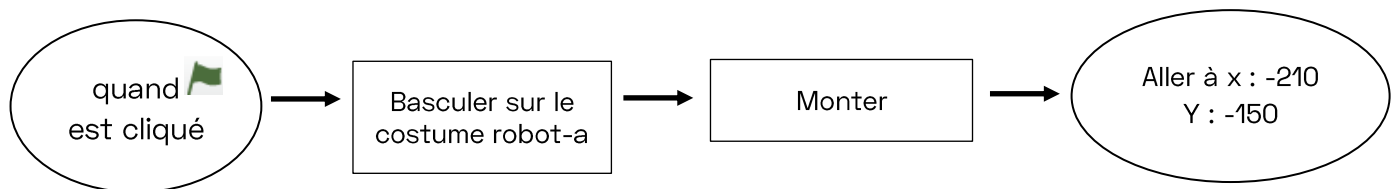


Ce bloc permet de déterminer où se place le robot dans la scène.

Les jeunes peuvent s'aider des codes couleur présents sur les blocs pour trouver la catégorie dans laquelle ils doivent chercher les blocs au sein de la palette d'instruction.

Une fois le code écrit dans la zone de programmation, ils sont invités à dessiner le logigramme de ce premier algorithme.

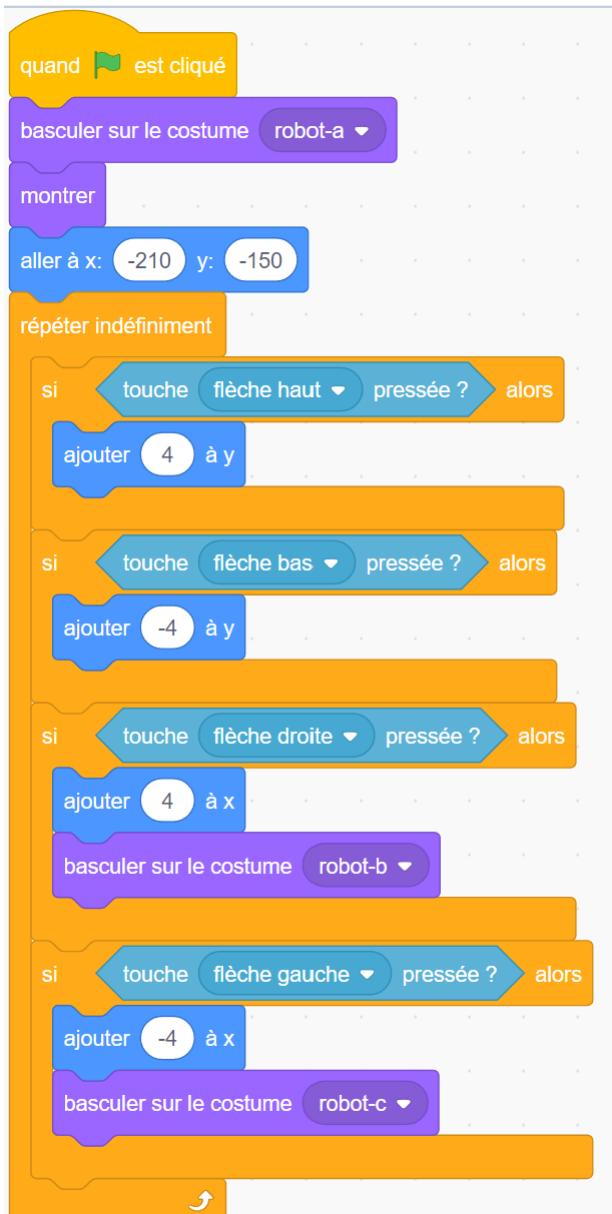
Logigramme:



Défi 4 – Contrôler le déplacement du Sprite

Pour permettre au robot de se déplacer dans la scène, les jeunes s'aident des touches fléchées présentes sur le clavier.

Voici les étapes à suivre :



1. Dans l'onglet "événements", choisir le bloc "quand le drapeau vert est cliqué".
2. Dans l'onglet "Apparence", sélectionner le bloc pour "basculer sur le costume robot a" et le bloc "montrer".
3. Dans l'onglet « mouvement », sélectionner un bloc "aller à x:0 y:0" pour placer le robot dans le coin en bas à gauche de la scène.
4. Dans l'onglet « contrôle », sélectionner un bloc "répéter indéfiniment" pour créer une boucle infinie.
5. Dans cette boucle, ajoutez une condition "si ..., alors" pour que "y soit augmenté de 4" si la touche "flèche du haut" est pressée (présente dans la catégorie capteurs).
6. Ajoutez une autre condition pour que "y soit diminué de 4" si la touche "flèche du bas" est pressée (présente dans la catégorie capteurs).
7. Ajoutez une autre condition pour que "x soit augmenté de 4" si la touche "flèche de droite" est pressée (présente dans la catégorie capteurs). Dans ce cas, faites basculer le robot sur le costume b.
8. Ajoutez une autre condition pour que "x soit diminué de 4" si la touche "flèche de gauche" est pressée. Dans ce cas, faites basculer le robot sur le costume c.



➔ Dans les deux dernières étapes, conscientisez les jeunes que lorsque le robot se déplace à gauche ou à droite, son apparence change. Il importe ainsi de basculer de costume selon le sens de direction.



À ce niveau, vous pouvez en profiter pour faire un rappel sur **les notions de boucle** (répéter indéfiniment) **et d'instruction conditionnelle** (Si... Alors).

→ Répéter indéfiniment : Le déplacement du robot en fonction des touches pressées doit se **répéter tout le long du jeu**. Dès lors, il est nécessaire d'utiliser une **boucle** qui se répète indéfiniment, pour que l'ensemble du code soit répété/exécuté en continu durant le jeu.

→ Si... alors : Dans la programmation, il est nécessaire de donner des instructions claires au programme. Cela se fait par exemple en imposant des **conditions** au robot pour qu'il exécute les actions que nous désirons. Nous employons donc le bloc « si... alors » pour indiquer le sens de déplacement à effectuer lorsque nous appuyons sur une touche pressée.

Il peut être utile d'insister sur la façon dont le robot va se déplacer : le code permet au robot de se déplacer grâce à des **changements de costumes** et des **opérations** sur les abscisses et les ordonnées du robot, plutôt que de le faire pivoter puis avancer. Cela permet de modifier les deux paramètres simultanément et d'avoir plus de liberté dans les mouvements.

Etape 3 : Mission test

Défi 1 – Tester le code

Les jeunes sont invités à tester leur code. Pour ce faire, ils cliquent sur le drapeau vert situé en haut à gauche de la scène.



Ils déplacent ensuite leur Sprite à l'aide des touches fléchées présentes sur leur clavier.



C'est l'occasion pour les jeunes de vérifier si le code fonctionne correctement et d'y apporter les modifications nécessaires.

Etape 4 : Déplacement du Sprite d'un point A à un point B

Pour tester le déplacement du robot que les élèves viennent de mettre en place, ils vont créer un petit parcours que le robot devra suivre, pour se rendre d'un **point A** à un **point B**.

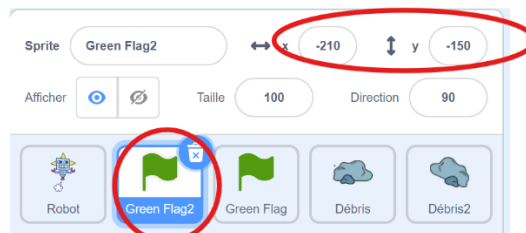


Défi 1 – Créer un parcours

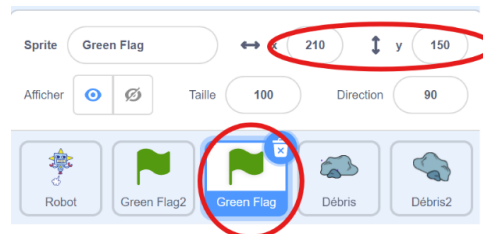
Les jeunes vont créer deux nouveaux Sprites. →  Ils vont les placer dans le **coin inférieur gauche** (**point A**) et le **coin supérieur droit** (**point B**) de la scène. Par exemple, ils peuvent prendre le Sprite "drapeau vert" présent dans la base de données Scratch.

Les jeunes peuvent directement indiquer les positions X et Y dans les cases de la zone Sprite ou bien en y placer les blocs dans la zone de programmation.

- Drapeau **coin inférieur gauche** de la scène (**point A**) : x : -210 , y : -150



- Drapeau **coin supérieur droit** de la scène (**point B**) : x : 210 , y : 150



Ils vont ensuite devoir positionner leur robot au **point A**, et le déplacer pour atteindre le **point B**. Ils peuvent tester plusieurs chemins différents pour essayer de sélectionner le chemin le plus court. Ils peuvent également essayer de se déplacer en longeant les bords du cadre.

Etape 5 - Évitement des débris spatiaux

Une fois que cela est fait, les jeunes peuvent ajouter des obstacles (nouveaux Sprites → ) à leur guise sur leur chemin qu'ils devront déplacer pour arriver du **point A** au **point B**.

Défi 1 – Créer des débris spatiaux

Les jeunes commencent par choisir un Sprite qui fera office de débris. Nous suggérons de sélectionner le Sprite « rocks » (rochers) présent dans la base de données de Scratch.



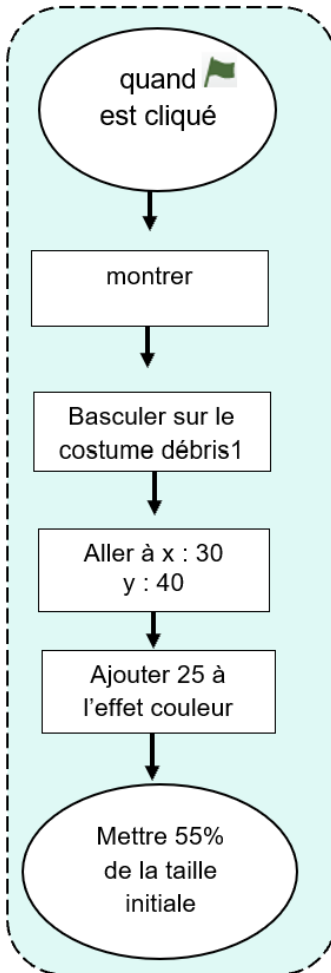
Défi 2 – Placer les débris spatiaux

Ils personnalisent ensuite chaque Sprite « débris » afin d'apporter de la variété dans les débris présents sur la scène. Pour ce faire, ils vont écrire l'algorithme de chaque débris sur base des logigrammes fournis.

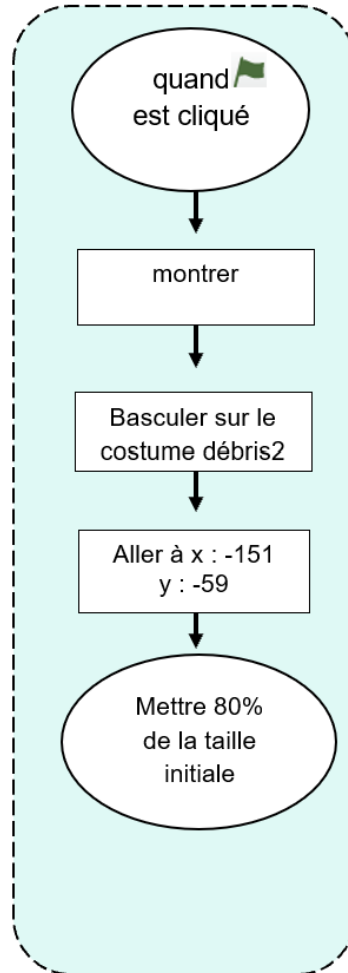
Les jeunes sont amenés à réaliser les deux premiers logigrammes donnés. Les logigrammes des débris 3 et 4 peuvent être donnés en tant que défis optionnels.

Voici le corrigé des logigrammes et des codes de chaque débris.

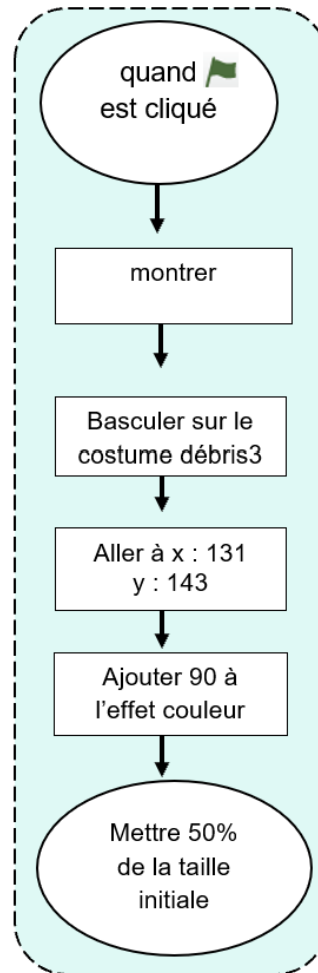
Logigramme débris 1



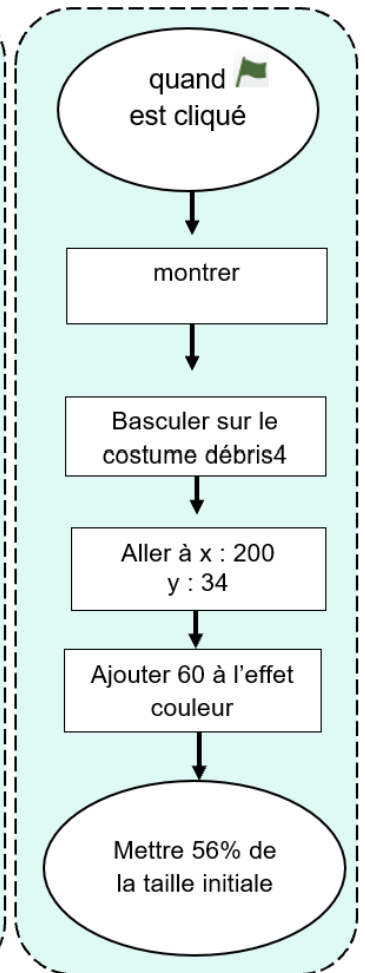
Logigramme débris 2



Logigramme débris 3



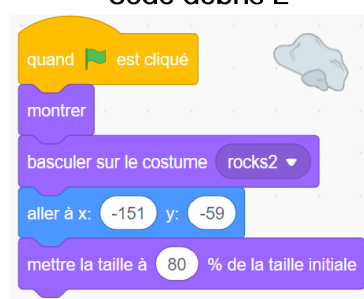
Logigramme débris 4



Code débris 1



Code débris 2



Code débris 3



Code débris 4



Défi 9 – Eviter les débris spatiaux

Le jeu est maintenant programmé : il est possible de le lancer et vérifier que tout fonctionne !

Les jeunes lancent leur programme et déplacent leur robot du drapeau **point A** au drapeau **point B** tout en évitant les débris spatiaux.

Pour voir le code final du programme du jeu, voici le lien : <https://scratch.mit.edu/projects/1212369710>

POUR ALLER PLUS LOIN

Il est possible d'ajouter un certain nombre d'options pour sophistiquer le jeu : éléments de décors, effets visuels, fonctions supplémentaires... Le [programme du jeu avec options](#) se trouve sur le site de Scratch.