


Scénario 4


Programmer avec Scratch Jr





3^e • 4^e

Programmer avec Scratch Junior

-  **Objectif du Plan d'études:** découvrir la science informatique en créant et exécutant des algorithmes et des programmes informatiques.

-  **Compétences opérationnelles, savoir-faire, savoirs:**
 - comprendre et utiliser les algorithmes et les programmes informatiques
 - exécuter/créer un algorithme simple
 - exécuter/créer un algorithme avec une boucle à une instruction
 - exécuter/créer un algorithme avec des boucles simples et des conditions simples
 - **savoir qu'un algorithme:**
 - est une succession d'étapes permettant de résoudre un problème, d'effectuer une tâche
 - peut contenir des instructions, des tests, des boucles
 - **savoir que:**
 - une boucle permet de répéter plusieurs fois les mêmes instructions
 - certaines boucles, dites *infinies*, ne s'arrêtent jamais

-  **Intentions pédagogiques:** afin de ne pas entraver la curiosité et la créativité des élèves, les laisser explorer et exploiter leurs découvertes comme ils·elles le souhaitent. Veiller cependant à ce que les notions proposées dans la progression soient comprises par tous les élèves à la fin de chaque palier. Au fil de l'avancée dans l'usage de Scratch Jr, passer en mode projet, avec définition du projet, découpage en étapes, réalisation, validation. Ceci permet de réinvestir les notions dans des situations plus abouties.

-  **Description de l'activité de base:** les élèves vont progressivement se familiariser avec une application de programmation. Ils·elles vont programmer des déplacements, découvrir les variables et les boucles à travers des activités simples et réinvestir leurs connaissances dans la réalisation d'un aquarium animé.

Séances	Résumé	Matériel
1: Découvrir Scratch Jr Durée: 45 minutes	Les élèves découvrent le logiciel Scratch Jr.	
2: Contrôler les déplacements Durée: 40 minutes	Les élèves programment sur Scratch Jr et apprennent à contrôler les déplacements des personnages.	
3: Utiliser des boucles Durée: 40 minutes	Les élèves programment sur Scratch Jr et créent une boucle.	<ul style="list-style-type: none"> • 1 vidéoprojecteur • 1 adaptateur VGA ou HDMI (tablette vers vidéo projecteur) • 1 tablette MASTER avec l'application Scratch Jr installée
4: Missions: utiliser des boucles Durée: 45 minutes	Les élèves relèvent de petits défis de programmation.	Pour les binômes: <ul style="list-style-type: none"> • 1 tablette pour 2 élèves avec Scratch Jr installée
5: Évaluation formative Durée: 45 minutes	Concevoir l'animation d'un aquarium.	

Notions Scratch Jr sur le Cycle 1

- Déplacements et fonctionnalités de base
- Glisser des blocs vers la zone de programmation
- Assembler des blocs
- Choisir un personnage
- Commencer le programme avec le drapeau vert
- Terminer le programme avec le bloc de fin rouge
- Choisir des arrière-plans
- Utiliser des boucles pour raccourcir la longueur d'un programme

Conseils pédagogiques

- Deux enfants par tablette: apprendre à programmer se fait en programmant, c'est pourquoi il est essentiel que chacun·e ait l'occasion de le faire. Nous conseillons de mettre les élèves par petits groupes devant les tablettes (idéalement, 2 élèves par tablette), et de confier la manipulation de celle-ci à un seul élève à la fois, avec changement des rôles toutes les 5 à 10 minutes.
- Tablette *master*: il est intéressant que l'enseignant·e ait une tablette de démonstration dont il·elle projette si possible l'écran. A défaut il·elle peut procéder par groupes. Cela servira d'une part pour montrer collectivement certaines manipulations, au rythme qui convient aux enfants; d'autre part, les corrections et mises en commun se feront également sur cette tablette-ci. Attention, selon la marque de la tablette, une telle opération peut être plus ou moins complexe et nécessiter des applications tierces (parfois payantes). Voir ici pour une synthèse: www.dane.ac-versailles.fr/enseigner-apprendre/innover/diffuser-l-ecran-d-une-tablette
- Fiche 2 à colorier en guise de trace écrite
- Tutorat dans la classe (si l'enseignant·e le souhaite): si certains groupes sont bloqués, un tutorat peut être mis en place et des élèves ayant trouvé une solution à leur problème peuvent venir montrer à d'autres ce qu'ils·elles ont découvert. Le tutorat doit être explicité aux élèves en amont de la séance. Cela permet à l'enseignant·e de faciliter la gestion de la séance, à l'élève tuteur d'aider et donc de mieux comprendre sans jamais donner la réponse et au tuteur de comprendre en recevant une aide ponctuelle.

Séance 1

Découvrir Scratch Jr

 **Résumé:** les élèves découvrent le logiciel Scratch Jr.

 **Matériel:**

- 1 vidéoprojecteur, 1 adaptateur VGA ou HDMI (tablette vers vidéoprojecteur), 1 tablette MASTER avec l'application Scratch Jr installée
- pour les binômes: 1 tablette pour 2 élèves avec Scratch Jr installée

Temps 1.1: Situation déclenchante

Modalités de travail: en grand groupe

 **Durée:** 5 minutes

Consigne: *Aujourd'hui, nous allons essayer, grâce à une application installée sur la tablette, d'animer un personnage. Vous allez devoir programmer en utilisant un langage spécial: un langage de programmation.*

L'enseignant·e explique aux élèves que ce langage est compréhensible à la fois par les enfants et par la tablette. Le langage et l'application qu'ils·elles vont utiliser s'appelle Scratch Junior. Avant cette séance, les élèves auront réalisé des activités débranchées (*jeu du robot idiot* par exemple) leur permettant de comprendre ce qu'est un langage de programmation.

Temps 1.2: Présentation de l'interface du logiciel Scratch Jr au groupe classe

Modalités de travail: en grand groupe


 **Durée:** 10 minutes



L'enseignant·e projette l'écran de la tablette de démonstration pour montrer aux enfants quelques manipulations essentielles. Il·elle doit montrer en grand groupe l'interface de Scratch Jr en mettant en avant les différents éléments qui la composent.

Consigne: *Pour démarrer le logiciel Scratch Jr, on clique du doigt une fois sur l'icône suivante, dans la liste des applications installées sur la tablette.*

On arrive alors sur un écran d'accueil qui comporte deux icônes: une maison et un point d'interrogation. Si on clique sur la maison, on accède à une nouvelle page, qui contient tout ce qui a déjà été créé sur cette tablette avec *Scratch Jr*.

Il faut cliquer sur l'icône  ci-contre pour créer un nouveau programme (*une nouvelle histoire*).



On accède alors à l'écran ci-contre (également présenté sur la Fiche 2). C'est à partir de cet écran que les élèves vont programmer la tablette.

On trouve sur cet écran :

1. les personnages ou lutins
2. la scène principale
3. l'ensemble des scènes
4. l'espace de programmation

Avec les différents blocs :

5. événements
6. mouvements
7. apparences
8. sons
9. contrôles
10. fins de programme



Sur la Fiche 1, vous trouverez les différentes briques (blocs) qui permettent de réaliser un affichage avec les élèves (affichage collectif et/ou un document individuel pour le cahier de l'élève). Cela permettra aux élèves de s'y référer pendant les activités de programmation et de fixer du vocabulaire commun lié au logiciel (le fond, la grille, les lutins, les décors...). Les lutins sont des objets manipulables (Sprite en anglais) que vous pouvez placer sur la scène centrale.

Temps 1.3: Découverte libre de Scratch Jr

Modalités de travail: par petits groupes, idéalement en binômes

 **Durée:** 20 minutes

L'enseignant·e laisse ensuite suffisamment de temps d'exploration de l'environnement pour que tous les élèves puissent tester l'effet des instructions de déplacement du chat (par exemple, 2 fois 10 minutes si les élèves forment des binômes).

Il est très important de changer les rôles afin que chaque élève puisse *manipuler* le logiciel: déplacements vers le haut, le bas, la droite et la gauche, rotation dans les deux sens, saut et retour à la position de départ, combinaison de plusieurs instructions de déplacement. On peut leur proposer de produire leur propre parcours du lutin.

Temps 1.4: Bilan collectif

Modalités de travail: en groupe classe

 **Durée:** 10 minutes

Après la découverte libre, regrouper les élèves et les tablettes et montrer les réalisations des groupes. On peut montrer les icônes (préalablement imprimées) qui ont été utilisées et établir un premier dictionnaire de la classe en nommant les blocs. Les élèves peuvent colorier les blocs qu'ils·elles ont utilisés dans leur programme via la fiche de suivi (Fiche 2).

Une affiche récapitulative peut être réalisée en précisant ce qu'est Scratch Jr (une application de programmation), ce qu'elle permet de faire, l'interface légendée (telle que présentée en phase 1 avec des étiquettes à replacer par exemple) et les quelques blocs vus lors du bilan avec leur fonction.

Exemple d’affiche à réaliser avec les élèves

Scratch Junior est une application de programmation



Les personnages ou lutins

La scène principale



L'ensemble des scènes

Blocs de programmation

Espace de programmation


On peut, par exemple programmer les déplacements de personnages, leur faire faire des actions (sauter, tourner, ...)


Quelques blocs de programmation

-  • Démarrer avec le drapeau vert
-  • Démarrer lorsqu'on touche le personnage
-  • Déplacer vers la droite
-  • Retourner à la position de départ
-  • Dire
-  • Fin
-  • Répéter indéfiniment

Séance 2

Contrôler les déplacements

 **Résumé:** les élèves programment sur Scratch Jr et apprennent à contrôler les déplacements des personnages.

 **Matériel:**

- 1 vidéoprojecteur, 1 adaptateur VGA ou HDMI (tablette vers vidéoprojecteur), 1 tablette MASTER avec l'application Scratch Jr installée
- pour les binômes: 1 tablette pour 2 élèves avec Scratch Jr installée

Temps 2.1: Rappel de la séance précédente

Modalités de travail: en groupe classe

 **Durée:** 5 minutes

Consigne: *La dernière fois, nous avons utilisé une application sur la tablette pour animer un petit personnage. Comment avez-vous fait? Qu'avez-vous utilisé? À quoi servent les différents blocs?*

En s'aidant de l'affiche réalisée, de la trace écrite individuelle et des souvenirs des élèves, l'enseignant·e effectue un rappel des activités de la séance précédente.


Consigne: *Aujourd'hui, nous allons apprendre à mieux contrôler les déplacements de notre personnage.*

Temps 2.2: Défis: Contrôler finement les déplacements du chat

Modalités de travail: en groupe classe puis en petits groupes

 **Durée:** 15 minutes



Avec le groupe classe et l'aide du vidéoprojecteur, l'enseignant·e fait apparaître la grille (icône ) et montre que lorsqu'on déplace le chat avec le doigt, une case bleutée se déplace avec lui.

Cette case indique la position du chat: par exemple, le chat est ici dans la case de la grille située au croisement de la troisième ligne en partant du bas et de la cinquième colonne en partant de la gauche, ce que l'on notera (ligne 3, colonne 5). L'enseignant·e demande aux élèves d'afficher la grille sur leur tablette et leur donne quelques défis de difficulté croissante.

- **Exercice A:** faire déplacer le chat de 3 cases vers la droite puis de 2 cases vers le haut.
- **Exercice B:** faire monter le chat de 3 cases vers le haut puis 3 cases vers le bas (revenir à la situation initiale).
- **Exercice C:** faire traverser l'écran de gauche à droite au chat, avec un saut tous les 4 pas (départ du chat de la colonne 3).
- **Exercice D:** faire faire tout le tour de la scène au chat (départ du chat de la case ligne 3, colonne 3).

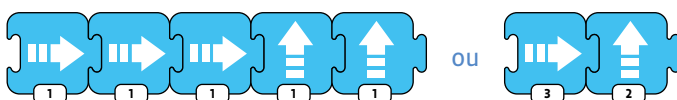


Pour chaque défi, un élève vient présenter sa solution en utilisant la tablette de démonstration. Certains élèves peuvent remarquer que le nombre 1 qui apparaît sur les instructions de déplacement peut être remplacé par un nombre plus grand, et qu'il contrôle le nombre de cases de déplacement du chat.

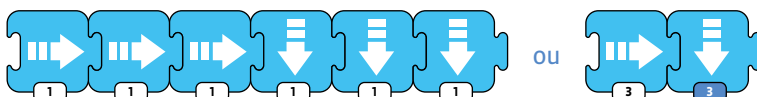
Dans ce cas, l'enseignant·e envoie l'un de ces élèves présenter sa solution au premier défi. Sinon, il·elle l'explique lui-même à la fin du premier défi : pour cela, il·elle peut faire coexister deux sous-programmes équivalents dans la zone de programmation et comparer leur effet en cliquant sur l'un ou sur l'autre. Il·elle demande aux élèves lequel ils·elles préfèrent. Il·elle précise que si l'on a beaucoup d'instructions à donner au personnage, il faut absolument privilégier le sous-programme qui prend le moins de place, et se débarrasser de l'autre en le glissant hors de la zone de programmation.

Les solutions des exercices seront donc :

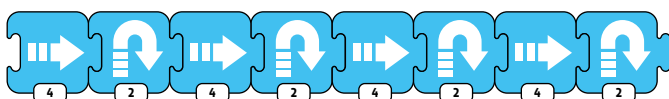
Exercice A :



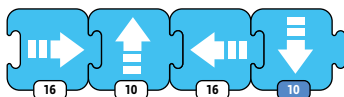
Exercice B :



Exercice C :



Exercice D :



Nota Bene


Si le chat atteint les limites de l'écran, on peut le replacer à une position de son choix par un cliquer-glisser.




Temps 2.3: Des événements pour déclencher les actions

Modalités de travail: en groupe classe

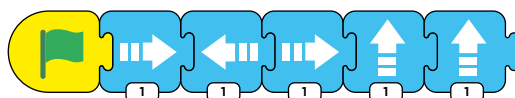
 **Durée:** 15 minutes

Consigne: *Je vais maintenant afficher la scène sur tout l'écran. Que se passe-t-il? Comment faire pour lancer les programmes?*

L'enseignant·e montre aux élèves, toujours à l'aide du vidéoprojecteur, comment passer en mode plein écran (icône , barre du haut). La classe constate alors deux choses: d'une part la grille disparaît (elle sert uniquement à faciliter la programmation des déplacements), d'autre part on ne peut plus lancer l'exécution des sous-programmes car ils n'apparaissent plus et on ne peut donc plus cliquer dessus. Un bouton reste disponible, même s'il n'a pas encore été utilisé jusqu'ici: le drapeau vert. Quand on clique sur ce drapeau vert, le chat revient à sa position de départ, mais il n'effectue pas les déplacements.

Ce problème se résout de la façon suivante: l'enseignant·e montre comment sortir du mode plein écran (clic sur l'icône , puis il·elle clique sur l'icône jaune  pour rendre les instructions de contrôle accessibles et choisit le bloc qui représente un drapeau vert: .

Puis, dans la zone de programmation, il·elle emboîte cette instruction juste à gauche des instructions que ce bloc doit contrôler, comme ceci dans le cas de l'exercice A:



Dorénavant, un *clic sur le drapeau vert* (appelé événement) déclenche le déplacement. Les élèves ajoutent l'instruction de contrôle à leur programme et testent son effet en plein écran. L'enseignant·e fait remarquer que l'exécution du programme donne toujours le même résultat.

Consigne: *Vous allez maintenant tester un autre bloc et voir à quoi il peut servir.*

L'enseignant·e demande alors aux élèves d'explorer par eux-mêmes la signification d'un autre événement, représenté par l'icône ci-contre. Les enfants produisent un sous-programme dont la première instruction est cette icône. Ils·elles cherchent comment déclencher ce sous-programme sans cliquer directement sur le bloc d'instructions.



Temps 2.4: Bilan et trace écrite

Modalités de travail: en groupe classe

 **Durée:** 5 minutes

Lors de la mise en commun, les enfants partagent leur conclusion: l'événement représenté par le doigt qui touche un personnage est un *clic sur le personnage*.

L'enseignant·e précise que toutes les instructions de Scratch Jr sont dessinées pour être facilement compréhensibles, et qu'il ne faut pas hésiter à essayer des instructions pour voir l'effet qu'elles produisent.

L'enseignant·e montre collectivement comment enregistrer les programmes : on clique sur le coin en haut à gauche de l'écran sur la maison orange.



On saisit le nom du fichier dans la case blanche (imposer des noms de fichiers très simples, comme les initiales *EP1* du binôme d'élèves Elise et Pierre pour le programme d'essai n° 1, puis on valide avec l'icône ✓).

Synthèse des conclusions pour les enseignant·es

- On peut donner des instructions à une machine en utilisant un langage spécial, appelé langage de programmation.
- On rassemble les instructions dans un programme, pour que la machine les applique.
- Si on lance le même programme plusieurs fois, il donne toujours le même résultat.
- Dans Scratch Jr, les programmes contiennent un ou plusieurs blocs d'instructions, appelés sous-programmes. Les sous-programmes sont déclenchés par des événements, comme *clic sur le drapeau* ou *clic sur un personnage*.

Pour les élèves, voici un exemple de trace écrite :

- on peut donner des ordres (des instructions) au lutin en le programmant avec des blocs.
- plusieurs blocs forment un programme.
- si on lance (on exécute) le programme plusieurs fois, cela donne le même résultat.

Puis les élèves complètent la Fiche 2: ils·elles colorient avec la couleur adéquate les instructions qu'ils·elles ont découvertes lors de cette séance et ajoutent quelques mots : plein-écran, grille, drapeau de démarrage, zone de programmation, instructions de déplacement, retour à la position de départ, rotation, saut, événements.

L'importance de l'écrit doit être soulignée, que ce soit dans la copie de la trace écrite, la définition du vocabulaire employé, les synthèses construites avec les élèves, les histoires imaginées, les affichages de la classe...

Séance 3

Utiliser des boucles

 **Résumé:** les élèves programment sur Scratch Jr et créent une boucle.

 **Matériel:**

- 1 vidéoprojecteur, 1 adaptateur VGA ou HDMI (tablette vers vidéoprojecteur), 1 tablette MASTER avec l'application Scratch Jr installée
- pour les binômes: 1 tablette pour 2 élèves avec Scratch Jr installée

Temps 3.1: Situation déclenchante

Modalités de travail: en grand groupe

 **Durée:** 5 minutes

Consigne: *Je vais vous raconter une petite histoire: Un jour de beau temps, un chat poursuivait un papillon blanc dans une belle forêt. Au détour d'un arbre, il aperçut une belle rivière dans une clairière verdoyante. Il avança vers un rocher près de la rivière. Puis, oubliant le papillon, il décida de remonter la rivière vers sa source. Je vous propose d'illustrer cette histoire avec notre application Scratch Jr. Nous allons animer notre personnage, le petit chat, pour qu'il se promène le long de la rivière.*


L'enseignant·e propose de changer la scène où se promène le chat pour commencer à raconter des histoires. Ici, il s'agira d'animer une saynète où le chat se promène le long d'une rivière.

La mission des élèves est la suivante: ils·elles doivent d'abord trouver la scène avec une clairière traversée par une rivière. Ensuite, ils·elles doivent placer le chat à côté de la forêt (près du papillon blanc), et contrôler ses déplacements pour qu'il se rapproche du rocher puis longe la rivière pour remonter le courant. Le héros doit commencer par suivre un chemin de type *petits escaliers* (ceci est imposé dans l'optique d'introduire les boucles en fin de séance).

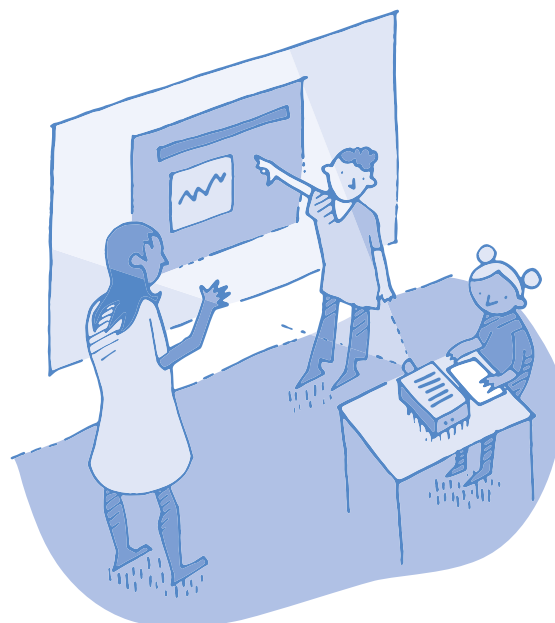
Temps 3.2: Choix de la scène

Modalités de travail: en grand groupe et en petits groupes

 **Durée:** 10 minutes

L'enseignant·e montre avec le vidéoprojecteur comment on accède aux différentes scènes proposées par l'application. Il faut cliquer sur l'icône  tout en haut de l'écran, puis on sélectionne et on valide une scène comme précédemment pour le personnage.

Consigne: *Vous allez essayer de trouver le décor qui correspond le mieux à la petite histoire que je viens de vous raconter.*



Consigne: *Je vous rappelle la situation de départ. Le chat sort de la forêt et poursuit un papillon blanc. Vous allez donc le placer en (colonne 2 ; ligne 6).*

Même en affichant un décor, la grille est toujours disponible afin d'aider les élèves à positionner correctement le personnage.

Une fois le chat placé au point de départ, l'enseignant·e peut préciser que les élèves vont faire des essais de déplacement, et qu'il·elle peut donc être intéressant d'avoir une instruction qui remplace automatiquement le personnage au point de départ.



Consigne: *Quel est, à votre avis, le bloc qui permet au chat de revenir à son point de départ ?*



Avec cette instruction, le personnage revient à son emplacement de départ.

Temps 3.3: Déplacement du chat

Modalités de travail: en grand groupe et en petits groupes

 **Durée:** 15 minutes

Consigne: *Vous allez maintenant écrire le programme pour que le chat se déplace jusqu'au rocher, puis qu'il remonte le long de la rivière.*

Pour que le chat se déplace à côté du rocher près de la rivière, on peut utiliser par exemple le programme ci-contre:



Pour que le personnage longe la rivière en remontant vers la source (en haut de l'image), on doit combiner plusieurs instructions de déplacement dans la zone de programmation. Selon le choix du personnage, sa taille et son point de départ, les instructions diffèrent légèrement.

Considérons, par exemple, que le chat démarre de la case (ligne 7, colonne 14):



Une fois les recherches effectuées, on invitera quelques groupes à réaliser leurs programmes devant le reste de la classe et on comparera les stratégies et les résultats obtenus.

Temps 3.4: Raccourcir le programme

Modalités de travail: en grand groupe et en petits groupes

 **Durée:** 10 minutes

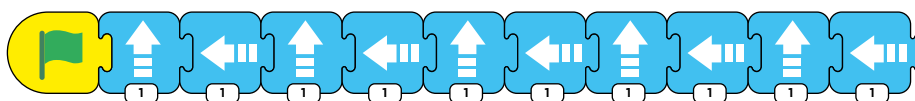
Lors de la présentation des programmes, quelques élèves auront probablement trouvé un moyen d'en obtenir un peu plus concis. On pourra comparer la longueur des programmes et mettre en avant ceux qui sont plus courts.

Pour la première partie de l'histoire, on peut raccourcir la longueur du programme en jouant sur les variables des déplacements. On passe ainsi de 15 instructions à 4 instructions.

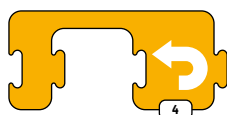
Pour la deuxième partie de l'histoire (remonter le cours de la rivière vers la source), c'est le bloc de couleur orange qui va servir à raccourcir la longueur du programme. Il est possible que certains élèves aient exploré les blocs de couleur orange, et localisé par eux-mêmes les *boucles*. Quoi qu'il en soit, l'enseignant·e introduit lui-même maintenant ce nouveau menu.




Consigne: *Pour raccourcir ce programme, on ne peut pas faire comme avec le premier. On va donc utiliser un nouveau bloc de programmation.*




Le programme ci-dessus comporte seulement 11 instructions élémentaires (en comptant l'événement de départ), mais il prend déjà beaucoup de place dans la zone de programmation. Si on veut raconter une histoire plus complexe, le programme risque de devenir long et compliqué. L'enseignant·e attire donc l'attention sur les séquences d'instructions qui se répètent (monter une fois, aller à gauche une fois).

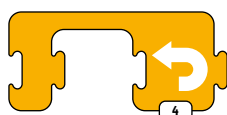


On compte combien de fois a lieu cette séquence. Et on introduit un nouveau bloc qui est accessible une fois qu'on a cliqué sur l'icône .

Les élèves vont devoir expérimenter pour comprendre comment elle s'utilise.

Pour tester ces blocs orange, il faudra sauvegarder le programme actuel pour recommencer des programmes vierges. L'enseignant·e rappelle comment faire et impose un nom de fichier très simple, comme **EP1** pour le premier programme concernant une histoire (rappel : les programmes d'essai ou les exercices du groupe sont nommés en fonction des initiales des prénoms des élèves), puis on valide avec l'icône .

Les élèves pourront colorier les nouveaux blocs utilisés sur leur fiche (Fiche 2). La trace écrite pourra être complétée de la manière suivante:



- Il est parfois nécessaire de raccourcir un programme trop long.
- On peut utiliser une boucle quand une ou plusieurs instructions se répètent.



Séance 4

Missions : utiliser des boucles

 **Résumé:** les élèves relèvent de petits défis de programmation.



Matériel:

- 1 vidéoprojecteur, 1 adaptateur VGA ou HDMI (tablette vers vidéoprojecteur), 1 tablette MASTER avec l'application Scratch Jr installée
- pour les binômes: 1 tablette pour 2 élèves avec Scratch Jr installée

Temps 4.1: Situation déclenchante

Modalités de travail: en groupe classe



Durée: 5 minutes

Consigne: *Aujourd'hui, avec Scratch Jr, vous allez réaliser de petits défis et utiliser tout ce que l'on a vu jusqu'à présent.*

Il est peut être nécessaire, avant de présenter les défis, de revenir sur les traces écrites, individuelles et/ou collectives. On pourra par exemple leur présenter des cartes avec des blocs d'instruction et interroger les élèves sur leur signification.

Temps 4.2: Mission 1

Modalités de travail: en petits groupes, puis en grand groupe

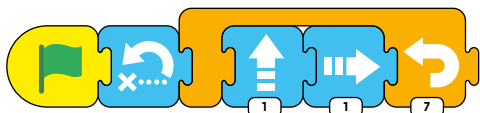


Durée: 10 minutes

Consigne: *Pour ce premier défi, vous allez devoir faire progresser le chat en escalier, c'est à dire un pas vers la droite, un pas vers le haut, et ainsi de suite. Il doit démarrer à l'emplacement (ligne 3, colonne 3).*

Dans un nouveau programme, les élèves vont devoir réinvestir les boucles afin que le chat répète la séquence d'instruction 1 pas à droite, 1 pas en haut, autant de fois que nécessaire. Le chat devra démarrer quand on cliquera sur le drapeau vert. Les élèves ne doivent pas oublier l'instruction *revenir à son point de départ*.

La mise en commun permettra aux élèves de présenter des programmes de ce type:



Temps 4.3: Mission 2

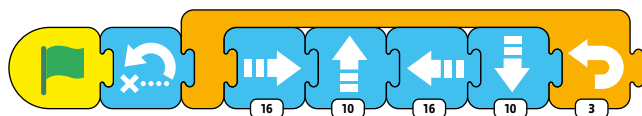
Modalités de travail: en petits groupes, puis en grand groupe

 **Durée:** 10 minutes

Consigne: Pour ce deuxième défi, vous allez devoir faire faire tout le tour de la scène au chat, 3 fois de suite. Il doit, encore une fois, démarrer à l'emplacement (ligne 3, colonne 3).

Là encore, les élèves devront utiliser une boucle. Ils-elles devront cependant saisir une variable pour chaque déplacement (à droite, en haut, à gauche, en bas) afin que le chat colle aux bords de l'écran. Ces variables pourront faire l'objet d'ajustements au fur et à mesure des essais et pourront être légèrement différentes d'un groupe à l'autre.

La mise en commun valorisera des programmes de ce type:



Temps 4.4: Mission 3

Modalités de travail: en petits groupes, puis en grand groupe

 **Durée:** 10 minutes

Consigne: Pour ce troisième défi, vous allez là encore faire faire le tour de la scène à votre personnage. Mais il existe une instruction qui permet de le faire indéfiniment, c'est à dire sans s'arrêter. À vous de trouver ce bloc et de voir comment on l'utilise.

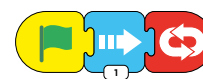


Les élèves vont devoir chercher parmi tous les blocs celui qui correspond à l'instruction *répéter indéfiniment*.


Ce bloc se trouve dans le menu rouge (encadré avec les blocs rouges):



La mise en commun permettra d'expliciter son utilisation puisque ce bloc ne vient pas encadrer des instructions mais se positionner à la fin d'une séquence d'instructions pour que l'ensemble du programme se répète à l'infini.



À l'issue de ce défi, on pourra compléter la trace écrite des élèves de la manière suivante:

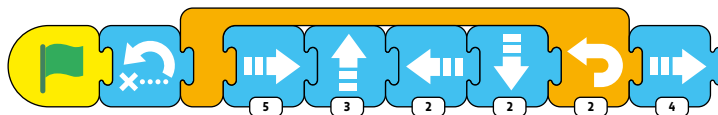
Quand on utilise le bloc , cela permet de répéter une boucle indéfiniment, c'est-à-dire sans s'arrêter.

Temps 4.5: Mission 4

Modalités de travail: en individuel.

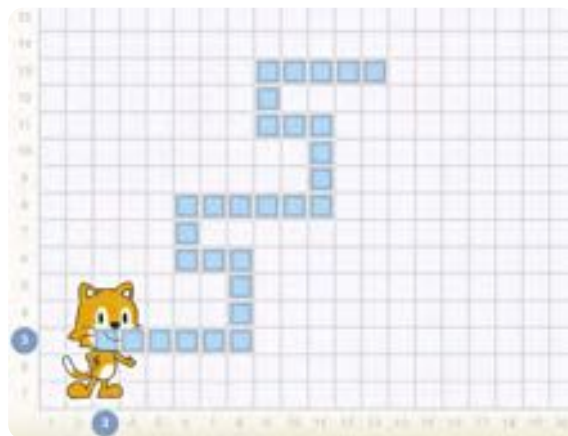
 **Durée:** 10 minutes

L'enseignant·e distribue à chaque élève la Fiche 3, et projette au tableau un programme simple combinant des boucles et des instructions de déplacement:



Consigne: Pour ce dernier défi, vous allez devoir dessiner, sur la fiche, les déplacements du chat qui correspondent à ce programme. Vous devez vous demander ce que va faire le personnage si on lance le programme.

Les élèves, individuellement, doivent ensuite tracer les déplacements du personnage sur la grille de la Fiche 3 (en coloriant les cases). Lors de la mise en commun, les élèves confrontent leurs propositions (notamment concernant la ligne et la colonne d'arrivée), et on vérifie collectivement en lançant effectivement le programme. Le trajet suivi par le chat est le suivant.



Séance 5

Évaluation formative : animer un aquarium

 **Résumé :** les élèves réalisent un aquarium animé.



Matériel :

- 1 vidéoprojecteur, 1 adaptateur VGA ou HDMI (tablette vers vidéoprojecteur), 1 tablette MASTER avec l'application Scratch Jr installée
- pour les binômes : 1 tablette pour 2 élèves avec Scratch Jr installée

Temps 5.1: Présentation du défi

Modalités de travail : par petits groupes



Durée : 5 minutes

Consigne : Aujourd'hui, avec Scratch Jr, vous allez animer un aquarium. C'est vous qui allez tout faire.

Vous allez devoir respecter quelques consignes :

- trouver une scène pour votre aquarium
- placer trois personnages :
 - un poisson qui va et vient en haut de la scène.
 - un crabe qui tourne en rond en bas de la scène.
 - un autre animal marin qui monte et qui descend.

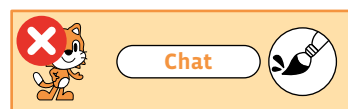
Dans un nouveau programme, les élèves vont devoir réinvestir les boucles afin que les différents personnages répètent les séquences d'instruction autant de fois que nécessaire. Pour cela, les élèves devront :

- choisir le décor de la scène (opter pour un fond marin),
- ajouter trois lutins *aquatiques* (poissons, crabe, hippocampe, pieuvre, étoile de mer, plongeurs : le choix est vaste),
- programmer des comportements différents en utilisant des boucles.

La nouveauté de cet exercice est la suppression du chat qui ne colle plus au thème de l'aquarium.

Si des élèves ont trouvé comment faire, ils-elles peuvent faire la démonstration grâce à la tablette Master, sinon l'enseignant·e introduit la méthode :

pour effacer le chat, il faut appuyer longtemps sur le rectangle *chat*, dans la zone en haut à gauche de l'écran (zone des personnages et objets), jusqu'à l'apparition d'une croix rouge (voir ci-dessus), puis cliquer sur cette croix.



Temps 5.2: Réalisation du défi

Modalités de travail : par petits groupes ou en individuel



Durée : 25 minutes

L'enseignant·e, en fonction du nombre d'élèves et de tablettes disponibles, peut choisir de faire travailler les élèves par 2 ou 3, ou bien s'assurer que chacun·e est capable de réaliser le défi et proposer cette activité individuellement, mais en plusieurs fois. La mise en commun se fera alors une fois que chaque élève aura réalisé son aquarium.

Temps 5.3: Mise en commun et blocs de couleur rose

Modalités de travail: en groupe classe et en petits groupes

 **Durée:** 10 minutes

L'enseignant·e pourra montrer successivement les réalisations des élèves à l'aide du vidéoprojecteur (mais peut-être pas toutes). Après cette projection, on pourra proposer d'ajouter un nouveau lutin, toujours aquatique. Mais ce lutin-ci devra uniquement utiliser les blocs de couleur rose. Les élèves devront explorer par eux-mêmes ce que ces blocs font.



En utilisant la tablette Master, des élèves présentent leurs trouvailles. Les blocs roses gèrent l'apparence des lutins : leur visibilité (affiché/masqué), leur taille, et même les bulles de leurs dialogues. Les premiers blocs seront les plus utiles (l'écriture de textes est assez peu pratique sur tablette, surtout pour des enfants à peine lecteurs), et les élèves terminent leur aquarium avec ces nouveaux blocs.

Conclusion

En conclusion de la séance, les élèves complètent la fiche d'utilisation de Scratch Jr, en coloriant les instructions rencontrées lors de cette séance et en ajoutant des mots-clés :

- bulle de texte écrit
- agrandir le personnage
- rétrécir le personnage
- rétablir la taille normale
- cacher le personnage (Masquer)
- afficher le personnage (Apparaître)



Bulle de
texte écrit



Agrandir le
personnage



Rétrécir le
personnage



Rétablir
la taille



Cacher le
personnage



Afficher le
personnage

Ressources et liens

- **Mission Scratch Jr (Atelier Canopé des Yvelines, France):**

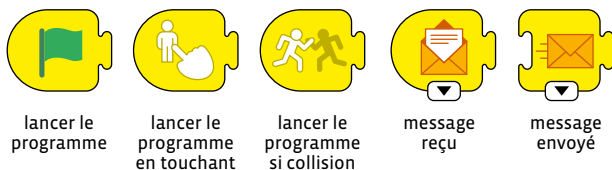
<https://www.reseau-canope.fr/atelier-yvelines/spip.php?article1161>

- **Scratch Jr, quels usages en classe (Atelier Canopé Besançon, France):**

<https://canope.ac-besancon.fr/codeetrobots/2018/06/scratch-junior-du-cycle-1-au-cycle-3-quelles-usages-en-classe/>

Le référentiel en français de Scratch Junior

Les briques de déclenchement des programmes



lancer le programme

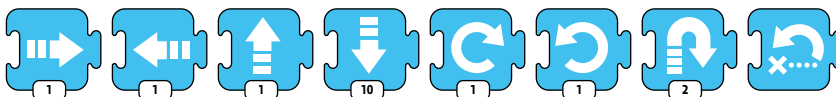
lancer le programme en touchant

lancer le programme si collision

message reçu

message envoyé

Les briques de mouvement des personnages



Aller à droite

Aller à gauche

Monter

Descendre

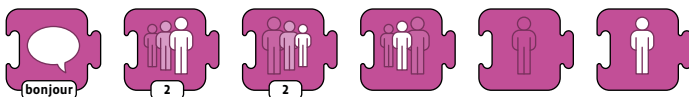
Tourner à droite

Tourner à gauche

Sauter

Position initiale

Les briques d'apparence des personnages



Bulle de texte écrit

Agrandir le personnage

Rétrécir le personnage

Rétablir la taille

Cacher le personnage

Afficher le personnage

Les briques pour les sons



Jouer le son "Pop"

Enregistrement n°1

Enregistrement d'un son

Les briques pour le contrôle



Attendre

Arrêt

Vitesse du personnage

Répéter

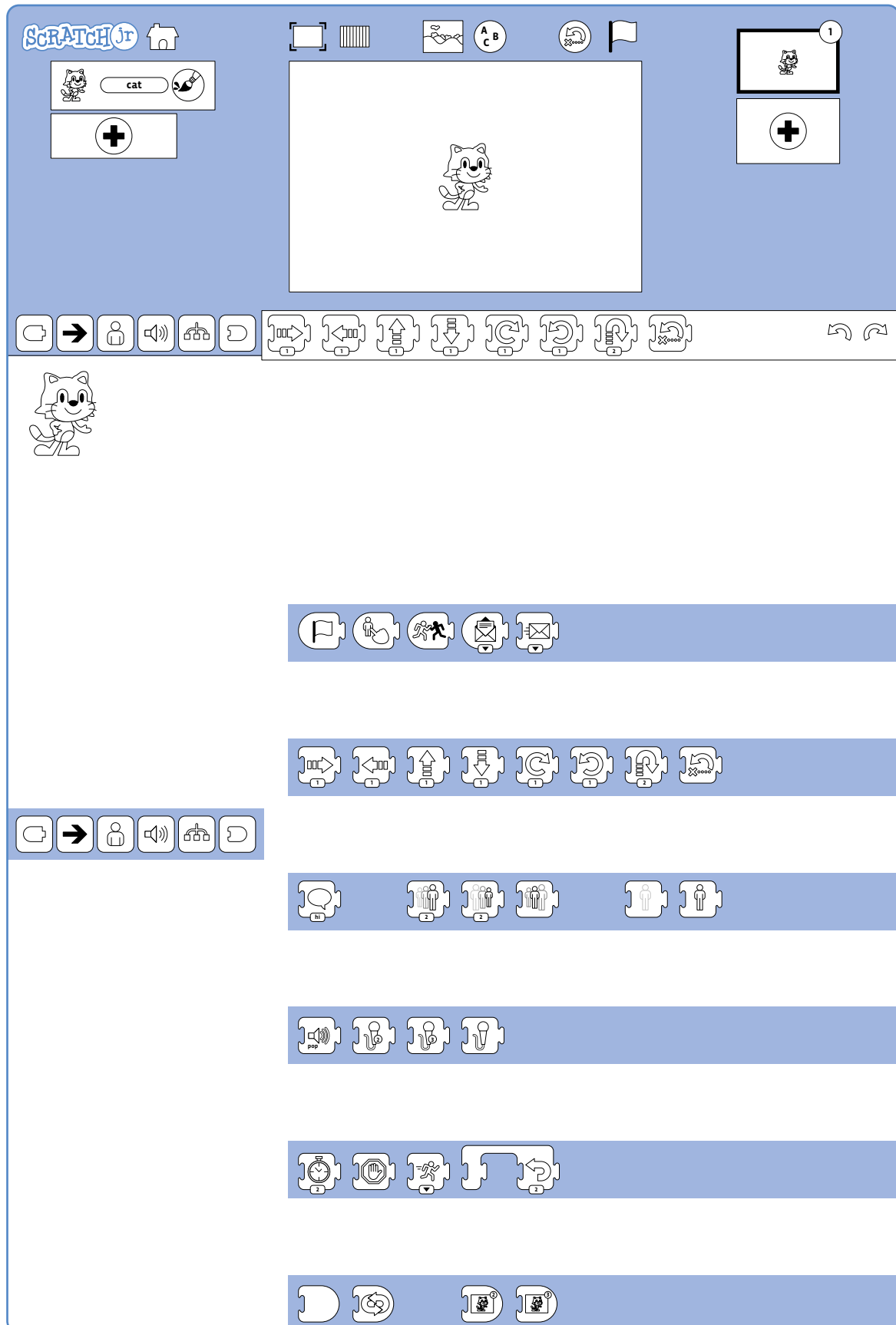
Les briques de fin de programmes



Fin

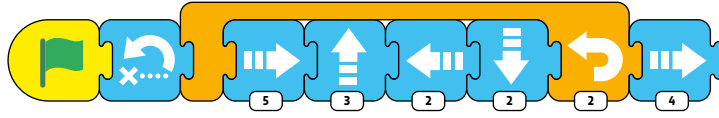
Toujours répéter

L'écran de programmation de Scratch Junior



Comprendre un programme avec une boucle

Consigne: le chat se trouve dans la case (ligne 3, colonne 3). Il suit les instructions du programme suivant:



Dessine tout son chemin sur la grille ci-dessous et entoure la case dans laquelle il termine son déplacement:

