

Programmer avec Scratch : algorithmique branchée (cycle 2)

Une séquence du projet *1,2,3... CODEZ !*

Résumé

Cette séquence est consacrée à la programmation d'une animation qui va raconter l'aventure vécue par le héros lors de la séquence débranchée « L'aventure ». Les élèves y découvrent Scratch et explorent les moyens de contrôler les mouvements d'un personnage. Ils racontent plusieurs passages de l'aventure de leur héros et apprennent ce faisant de nouvelles fonctionnalités, dont le chargement de nouvelles scènes ou l'ajout de lutins. Ils font alors en sorte que les deux premiers épisodes de l'aventure du héros s'enchaînent : pour cela, ils découvrent la notion de message. Ils approchent également la notion de boucle. Enfin, ils sont invités à produire un « épisode final », en autonomie, à titre de réinvestissement.



Séance 1 – Découverte de l'environnement de programmation *Scratch*

Résumé	Les élèves découvrent <i>Scratch</i> , un environnement de programmation graphique simple d'utilisation.
Notions (cf. scénario conceptuel, page 114)	<p>« Machines » et « Langages »</p> <ul style="list-style-type: none"> • On peut donner des instructions à une machine en utilisant un langage spécial, appelé « langage de programmation », compréhensible par l'homme et la machine. • Un « algorithme » est une méthode permettant de résoudre un problème. • Un programme est un algorithme exprimé dans un langage de programmation.
Matériel	<p>Pour la classe :</p> <ul style="list-style-type: none"> • (conseillé) Un ordinateur sur lequel le logiciel <i>Scratch</i> a été installé et un système de vidéo-projection <p>Pour chaque binôme :</p> <ul style="list-style-type: none"> • Un ordinateur sur lequel l'application <i>Scratch</i> a été installée (voir la note p. 171) <p>Pour chaque élève :</p> <ul style="list-style-type: none"> • Fiche 32, page 249 (cette fiche sera réutilisée dans les autres séances); l'exploiter (ou alors la distribuer séance 2)
Lexique	Programme, script, personnage, scène, instruction, événement
Durée	1 h

Avant-propos

Comme expliqué précédemment, cette séquence *Scratch* au cycle 2 reprend, lors des premières séances, certaines activités décrites très en détail dans la progression pour le cycle 3. Nous ne réécrivons pas tout ici et faisons de fréquents renvois aux séances cycle 3 concernées. Il est très fortement recommandé, avant de poursuivre, d'avoir lu au moins les trois premières séances de la séquence *Scratch* cycle 3 (et d'avoir fait, chez soi, les activités proposées).

Le scénario général de cette séquence 2bis, en revanche, est très similaire à celui de la séquence 2 pour le cycle 2, sauf que nous utilisons *Scratch* plutôt que *Scratch Junior*. Là aussi, nous faisons des renvois aux séances concernées plutôt que de tout redétailler ici.

Notes pédagogiques

- Apprendre à programmer se fait en programmant, pas en regardant quelqu'un programmer ! Il est intéressant de réfléchir à 2 sur un même problème (probablement plus que de programmer seul), mais il est important d'être actif. Nous conseillons donc de mettre les élèves par petits groupes devant les machines (idéalement, 2 élèves par machine) et de leur demander de « passer la main » (donner le clavier et la souris à son voisin) toutes les 10 ou 15 minutes.
- Nous conseillons, si possible, d'organiser la classe en demi-groupes de façon à ne pas avoir trop de binômes à gérer. Pendant que la moitié de la classe travaille sur *Scratch*, l'autre moitié fait autre chose, en autonomie.

- Si possible, faire 2 séances de *Scratch* par semaine, surtout au début du projet.
- Cette étape de découverte de *Scratch* est, volontairement, très directive (elle commence même par une démonstration de l'enseignant!). C'est la seule qui se présente sous cette forme. Tous les binômes doivent accomplir une série de tâches élémentaires. À la fin de chaque tâche, une mise en commun permet de s'assurer que chacun a compris et sait faire. Les autres étapes sont moins dirigées, les élèves devenant plus autonomes et avançant chacun à leur rythme.
- Pour gagner du temps, allumer les ordinateurs avant le début de la séance.

Situation déclenchante

L'enseignant explique aux enfants qu'ils vont utiliser un ordinateur pour raconter les principaux épisodes de l'histoire de leur héros. Pour cela, ils vont devoir programmer l'ordinateur, c'est-à-dire lui dire quoi faire. Il faudra utiliser un langage spécial, un langage de programmation, compréhensible à la fois par les enfants et par l'ordinateur. Le langage qu'ils vont utiliser s'appelle *Scratch*. Aujourd'hui et à la prochaine séance, ils vont découvrir *Scratch*, et à partir de la troisième séance, ils commenceront le récit de l'histoire.

Lancement de Scratch et découverte de son interface

Après une présentation rapide de *Scratch*, les élèves explorent librement le logiciel, puis exécutent des exercices simples. Voir l'étape 1 de la séquence *Scratch cycle 3* (page 241), en particulier les tâches suivantes :

- Tâche 1 : lancer *Scratch* et découvrir son interface (page 243). L'enseignant réalise une démonstration rapide de *Scratch* : présentation des différents éléments (scènes, lutins, arrière-plans, onglet « script ») et d'actions élémentaires (glisser des instructions dans la zone de programmation, combiner des instructions en sous-programmes, supprimer des instructions). L'enseignant distribue alors la Fiche 32, que les élèves colorieront au fur et à mesure de la séquence.
- Tâche 2 : explorer librement *Scratch* (page 245). Les élèves découvrent les différentes catégories d'instructions (« mouvement », « apparence », « événement », « contrôle »...)
- Tâche 3 : s'approprier l'interface en résolvant 7 petits exercices progressifs (page 246).

Conclusion et traces écrites

La classe synthétise collectivement ce qui a été appris au cours de cette séance :

- on peut donner des instructions à une machine en utilisant un langage spécial, appelé « langage de programmation », compréhensible par l'homme et la machine ;
- un « algorithme » est une méthode permettant de résoudre un problème ;
- un programme est un algorithme exprimé dans un langage de programmation.

Les élèves notent cette conclusion dans leur cahier de sciences et complètent la fiche d'utilisation de *Scratch* (Fiche 32) en coloriant les instructions rencontrées et en notant quelques mots-clés.



Séance 2 – Déplacer un personnage

Résumé	Les élèves explorent les moyens de contrôler les mouvements d'un personnage.
Notions (cf. scénario conceptuel, page 114)	Idem séance précédente
Matériel	Idem séance précédente En plus, pour chaque élève : Fiche 32, page 249
Lexique	
Durée	1 h

Lors de la séance précédente, les élèves ont commencé à s'approprier l'interface de *Scratch*. Ils vont maintenant apprendre comment diriger précisément les lutins, en s'entraînant sur le chat.

Note pédagogique

Pour cette étape, les élèves auront encore besoin d'être guidés. Ensuite, ils auront acquis les automatismes leur permettant d'être bien plus autonomes, et chaque binôme avancera à son rythme.

Défis : contrôler les déplacements du chat (idéalement par binômes)

Grâce aux exercices déjà réalisés, les élèves savent faire avancer le chat vers la droite (cf. tâche 1 page 244). Il va falloir le faire avancer dans n'importe quelle autre direction.

Dans un premier temps, les élèves doivent faire avancer le chat vers la gauche. Le faire avancer vers la gauche est une tâche un peu plus difficile, car les élèves doivent d'abord demander au chat de s'orienter vers la gauche, avant de lui demander d'avancer.

Ils travaillent en autonomie et tâtonnent, l'enseignant passant régulièrement dans les groupes pour s'assurer que personne n'est bloqué. Il peut les guider en les incitant à chercher une instruction « s'orienter à ».



Les élèves testent l'effet du bloc de deux instructions « s'orienter à... » et « avancer de 10 », selon que l'instruction « s'orienter à » indique 90, -90, 0 et 180.

Dans le même temps, l'enseignant encourage les élèves à cliquer sur l'icône d'information sur le lutin (lettre « i » dans un cercle bleu) et à tester les trois styles de rotation possibles.



Ainsi, on peut orienter le lutin en manipulant le tiret bleu « direction ». De plus, à tout moment, lorsque l'on utilise l'instruction « s'orienter à », ce tiret bleu se met à jour. On peut quitter le mode « information » en cliquant sur la petite flèche bleue « retour » dans le cercle bleu.

Les élèves trouvent alors comment relever le défi : en combinant les instructions « s'orienter à -90 » et « avancer de 10 ».



Armés de ces deux nouveaux outils, les élèves peuvent alors répondre définitivement à la question initiale : diriger le chat dans n'importe quelle direction.



Classe de CE2 d'Emmanuelle Wilgenbus (Antony)

Conclusion et traces écrites

Lors de cette séance, les élèves n'ont pas mobilisé de nouveaux concepts d'informatique, mais ont renforcé les acquis. En conséquence, ils complètent simplement leur fiche d'utilisation de *Scratch* (cf. séance précédente) : ils y colorient, dans la couleur adéquate, les instructions qu'ils ont découvertes lors de cette séance et ajoutent quelques mots-clés.



Séance 3 – Choisir le héros et contrôler ses déplacements

Résumé	Les élèves racontent un premier passage de l'aventure de leur héros, dans lequel, tout juste sorti de la forêt, il longe la rivière jusqu'à la mer. Ce faisant, ils réinvestissent des notions vues aux séances précédentes (séquence d'instructions et événement), découvrent la notion d'initialisation et utilisent des boucles.
Notions (cf. scénario conceptuel, page 114)	« Algorithmes » <ul style="list-style-type: none"> • Une boucle permet de répéter plusieurs fois la même instruction. • Certaines boucles sont répétées un nombre précis de fois.
Matériel	Idem séance précédente L'enseignant aura préparé les postes de travail en copiant les fichiers <i>Scratch</i> (extension .sb2) mis à disposition sur le site Web du projet (voir page 349)
Lexique	Initialisation, boucle « répéter ... fois »
Durée :	1 h

Situation déclenchante

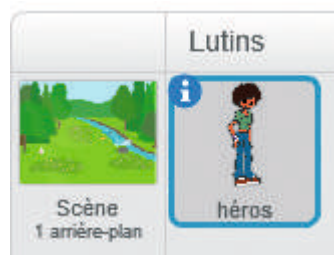
L'enseignant demande aux élèves de se remémorer collectivement les aventures de leur héros: il s'est retrouvé mystérieusement au sommet d'une montagne et a réussi à atteindre une clairière. Là, sur un tronc d'arbre, il a trouvé un message qu'il a réussi à décoder. Ce message l'incitait à suivre la rivière jusqu'à la mer et à rechercher un trésor au fond de l'eau. L'épisode que les enfants vont raconter aujourd'hui, avec *Scratch*, est celui où le héros longe la rivière.

Choix d'un héros et contrôle de ses déplacements (idéalement par binômes)

Les élèves ouvrent le fichier pré-rempli¹⁶ «II-2bis_riviere.sb2» à l'aide du logiciel *Scratch*. La scène représente une forêt, une clairière et une rivière, afin que les élèves puissent raconter l'épisode où le héros suit la rivière. Un lutin est déjà présent: le héros de l'histoire. Il est positionné vers la gauche de l'écran juste à la sortie de la forêt,

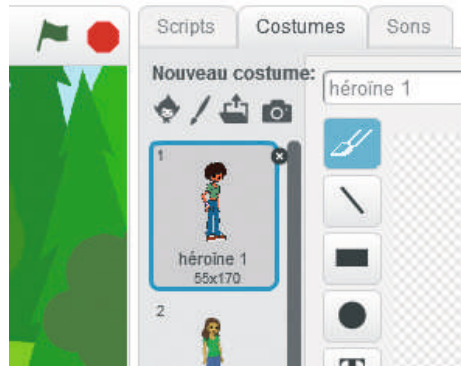
Les élèves commencent par choisir le personnage principal de l'histoire parmi 3 héroïnes et 3 héros possibles. Pour cela, ils doivent :

- sélectionner le lutin « héros » par un clic sur lui dans la zone « Lutins » :



16. Toutes les sources pour la programmation *Scratch Junior*, *Scratch* et *Thymio* sont disponibles sur le site Web du projet. Voir page 349.

- cliquer sur son onglet « Costumes » pour visualiser les 6 personnages possibles;

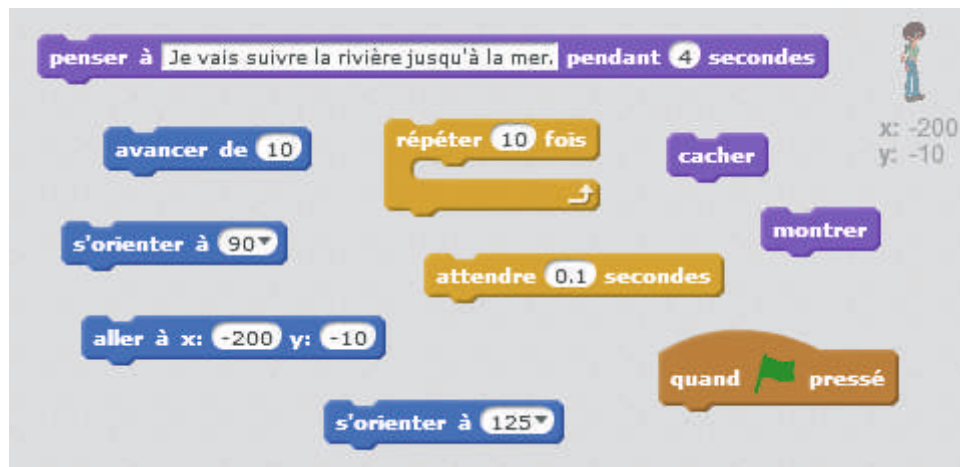


- Faire glisser le costume de leur choix en première position et éventuellement supprimer les costumes inutiles en cliquant sur la petite croix grise qui apparaît lorsqu'on les sélectionne.
- Revenir à la programmation en cliquant sur l'onglet « Scripts ».

Une fois ce travail effectué, les élèves ont pour mission de contrôler les déplacements du héros pour qu'il aille jusqu'à la rivière puis la longe jusqu'au coin en bas à droite de l'écran avant de disparaître. Les déplacements doivent être déclenchés par un clic sur le drapeau vert.

Les élèves doivent utiliser uniquement des instructions qui sont déjà dans la zone de programmation du personnage (éventuellement en plusieurs exemplaires, auquel cas, il leur faut retourner chercher des exemplaires de ces instructions dans la liste des instructions au centre de l'écran, ou les dupliquer grâce à un clic droit). Ils ne sont pas obligés d'utiliser toutes les instructions.

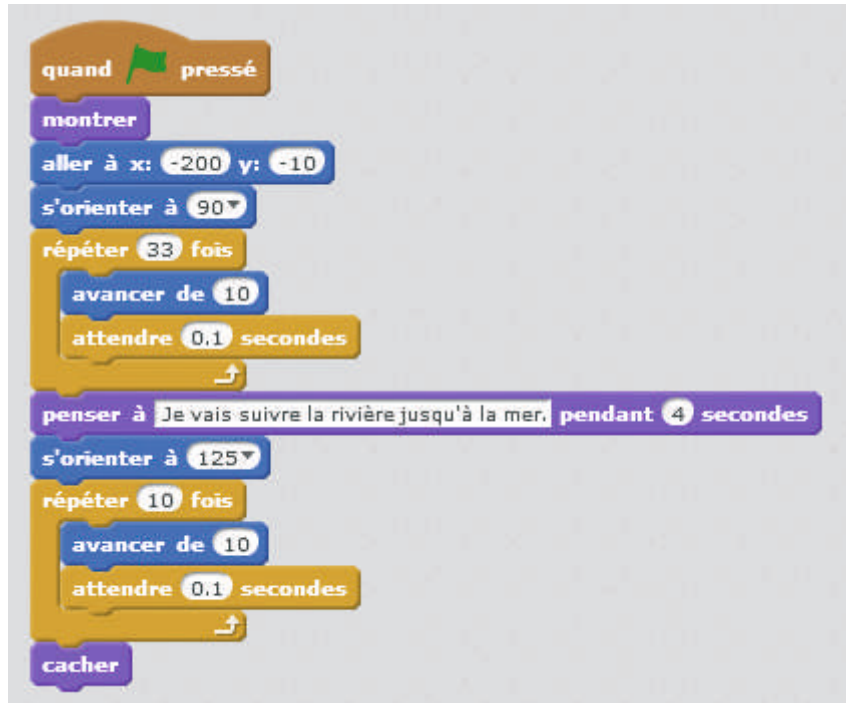
Les instructions utilisables sont les suivantes, et il est possible d'en modifier les valeurs :



Mise en commun (collectivement)

Un binôme vient présenter sa proposition à l'ordinateur de démonstration. La classe s'appuie sur cette présentation pour discuter des difficultés rencontrées par les différents binômes et partager les solutions trouvées.

Il peut y avoir autant de propositions que de groupes, mais une possibilité est la suivante :



Notes pédagogiques

- Nous avons choisi de pré-sélectionner 6 personnages plutôt que de laisser les élèves choisir dans la vaste bibliothèque de lutins de Scratch, afin que les personnages aient tous une dimension semblable, qu'ils aient tous la même orientation de départ et surtout afin que le temps de la séance soit affecté à de la programmation plutôt qu'à d'interminables discussions sur le choix d'un personnage. Il est toutefois possible, sur un temps d'arts plastiques ou de TICE, de laisser les élèves créer leur personnage à l'aide des outils de dessin de Scratch, ou de modifier un personnage existant.
- Au départ, les élèves imaginent le plus souvent de contrôler le déplacement du héros en une seule instruction, par exemple : « avancer de 400 ». Mais ils s'aperçoivent que le déplacement est quasiment instantané et pas du tout réaliste. L'idée émerge alors de faire plusieurs petits déplacements, en utilisant ou pas l'instruction « répéter ... fois ». Mais si les élèves n'utilisent pas l'instruction « attendre ... secondes » entre deux déplacements successifs, cela ne règle pas le problème.
- Les élèves pensent généralement à utiliser l'instruction « cacher » pour que le personnage disparaisse à la fin du programme, mais lorsqu'ils essaient de relancer le programme, leur personnage n'apparaît pas. Il conserve en effet son état « caché », car aucune instruction ne lui demande d'en changer. Ce problème se règle facilement en ajoutant une instruction « montrer » en début de programme. Cette instruction est une initialisation, qui donne la situation initiale, tout comme l'instruction « s'orienter à 90 ».
- De la même façon, si les élèves positionnent leur personnage à la lisière de la forêt par un cliquer-glisser sur la scène, sans imposer cette position initiale par une instruction dans le programme, le personnage part de cette position lors du premier lancement, mais n'y revient pas lorsque le programme est relancé. C'est ce qui justifie l'ajout de l'instruction « aller à x:... y:... » en début de programme. Cette instruction est aussi une initialisation. Remarquons que le positionnement manuel

du personnage a tout de même un intérêt: l'instruction «aller à x:... y:... » prend les valeurs de x et y correspondant à cette position et il n'y a plus, alors, qu'à placer cette instruction dans la zone de programmation pour s'en servir.

- L'enseignant oriente la discussion sur le rôle de l'instruction «répéter ... fois» et fait émerger le terme «boucle», vu à la Séance 5 (page 135) et à la Séance 6 (page 140) de la séquence précédente. Il demande où doivent être placées les instructions à répéter: elles doivent être prises comme en étau dans l'instruction «répéter ... fois». L'enseignant précise que dans ce type de boucle, on décide à l'avance combien de fois on veut répéter les instructions: le nombre de répétitions est défini à l'avance, ou prédéfini.

Les groupes ont la possibilité de modifier leur première proposition à l'issue de la mise en commun, par exemple pour utiliser une boucle s'ils avaient fait sans. L'enseignant montre comment enregistrer le programme (Fichier, Enregistrer sous, emplacement à préciser) et impose un nom: A1 ou B1 pour ce premier épisode, selon que les élèves sont de la demi-classe A ou B.

Le fichier «Il-2bis_riviere_correction_seance_3.sb2» fournit un exemple de ce que l'on obtient à l'issue de cette séance.

Conclusion et traces écrites

La classe synthétise collectivement ce qui a été appris au cours de cette séance:

- *Certaines instructions servent à préciser un état de départ, aussi appelé «état initial»: ce sont des instructions d'initialisation.*
- *Une boucle permet de répéter plusieurs fois la même instruction. Certaines boucles sont répétées un nombre précis de fois.*

Les élèves notent ces conclusions dans leur cahier de sciences. Puis ils complètent leur fiche d'utilisation de *Scratch*: ils y colorient dans la couleur adéquate les instructions qu'ils ont découvertes lors de cette séance et ajoutent quelques mots: boucle prédéfinie, pause.



Séance 4 – Programmer plusieurs lutins

Résumé	Les élèves doivent raconter un deuxième passage de l'aventure de leur héros, dans lequel il voit le trésor situé au fond de la mer et se fait aider pour le récupérer. Pour cela, les élèves apprennent à charger une nouvelle scène, à ajouter un lutin et réinvestissent les notions de programmation des deux premières séances.
Notions (cf. scénario conceptuel, page 114)	« Algorithmes » <ul style="list-style-type: none"> Des instructions peuvent démarrer en même temps, si leur exécution est déclenchée par un même événement.
Matériel	Idem séance précédente
Lexique	
Durée :	1 h

Situation déclenchante

L'enseignant demande aux élèves ce qui arrive au héros une fois qu'il a longé la rivière jusqu'à la mer. La classe rappelle que le héros voit le trésor au fond de la mer et qu'il dirige un sous-marin pour récupérer le trésor. L'enseignant propose de programmer cet épisode aujourd'hui.

Changer de scène et ajouter un lutin (collectivement puis par binômes)

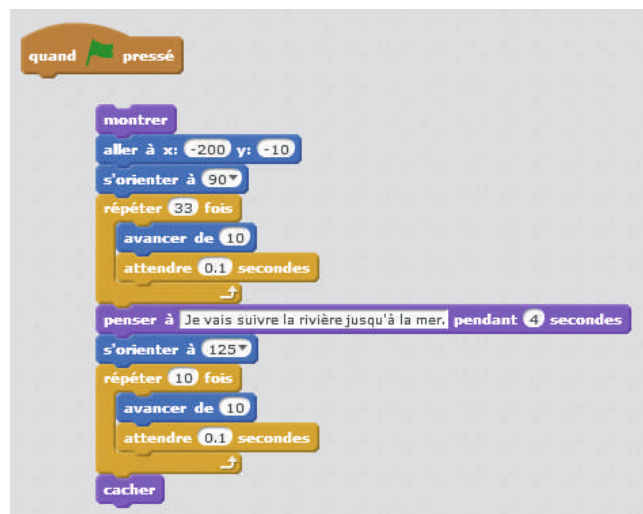
Tout d'abord, l'enseignant montre à la classe, à l'aide de l'ordinateur de démonstration, comment charger une nouvelle scène (voir la tâche 2 de l'étape 2 pour le cycle 3, page 250) : il faut cliquer sur la scène actuelle (la rivière), cliquer sur l'onglet « arrière-plans » et charger un arrière-plan à partir d'un fichier (le fichier d'arrière-plan que nous fournissons s'intitule « paysage_ponton »). Les élèves reproduisent le tout sur leur propre ordinateur.

L'enseignant montre ensuite comment charger un nouveau lutin depuis la bibliothèque *Scratch*. Comme il n'y a pas de sous-marin, il choisit par exemple une pieuvre, avec l'accord des élèves. Les élèves chargent eux-aussi le lutin « pieuvre ».

L'enseignant montre que la scène et chaque lutin ont chacun leur zone de programmation. Pour créer un programme pour un certain lutin (ou pour la scène), il faut commencer par sélectionner le lutin (ou la scène) en cliquant sur lui dans la zone des lutins (ou sur la scène).

L'enseignant explique enfin que pour le moment, on va inactiver le script du héros correspondant à l'épisode de la rivière : pour cela, il suffit de séparer tout le bloc d'instructions de l'événement déclencheur « Quand drapeau vert pressé », comme ceci :

On s'occupera plus tard (séance suivante) de remettre ces instructions en service. Les élèves procèdent à cette inactivation.



Programmer l'épisode (par binômes)

Il est temps pour les élèves de programmer le nouvel épisode : le héros doit se rendre tout au bout du ponton, où il rencontre la pieuvre faisant des allers-retours de droite à gauche. Lorsque le héros arrive au bout du ponton, la pieuvre s'arrête à proximité de lui. Après un court dialogue, la pieuvre disparaît (elle part chercher le trésor).

Note pédagogique

Nous proposons ici de laisser les élèves chercher par eux-mêmes les instructions, car la plupart ont déjà été utilisées à la séance précédente. Toutefois, en cas de trop grande difficulté, ne pas hésiter à rappeler les instructions utiles.

Mise en commun

Lors de la mise en commun, un groupe d'élèves vient présenter sa proposition. La classe s'appuie sur cette présentation pour discuter des difficultés rencontrées et des solutions trouvées.

Une solution possible est la suivante, sans les dialogues (à gauche le programme du héros, à droite le programme de la pieuvre, comme le montre la silhouette du lutin en haut à droite de la zone de programmation) :



Programme du héros

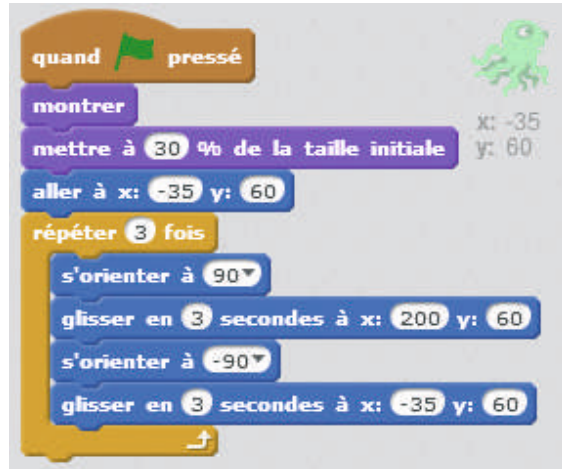


Programme de la pieuvre

Notes pédagogiques

- Peu d'élèves se préoccupent d'initialiser la position du lutin au démarrage du programme. Faire ré-émerger la notion si besoin : « Comment avons-nous fait pour que le héros soit toujours visible au début de l'épisode de la rivière ? », « Comment avons-nous fait pour que le héros soit toujours à la lisière de la forêt au début de l'épisode de la rivière ? » Il faut tout simplement préciser les valeurs des coordonnées X et Y souhaitées, à l'aide de l'instruction : « aller à x:... y:... »
- Peu d'élèves pensent à choisir le style de rotation de la pieuvre. Elle se retrouve alors la tête en bas lorsqu'elle se déplace vers la gauche.
- L'enseignant peut introduire l'instruction « glisser en... secondes à x:... y:... » pour simplifier le programme de la pieuvre, dont la version de la page 182 comporte des boucles imbriquées.

Cela donne par exemple :



Le fichier «II-2bis_ponton_correction_seance_4.sb2» fournit un exemple de ce que l'on obtient à l'issue de cette séance.

Conclusion et traces écrites

Aucune nouvelle notion d'informatique n'a été abordée au cours de cette séance, mais les élèves ont découvert de nouvelles instructions *Scratch*. Ils complètent donc leur fiche d'utilisation de *Scratch* en coloriant les instructions nouvelles et en ajoutant quelques mots : lutin sélectionné, boucle.



Séance 5 – Coordonner les deux premiers épisodes

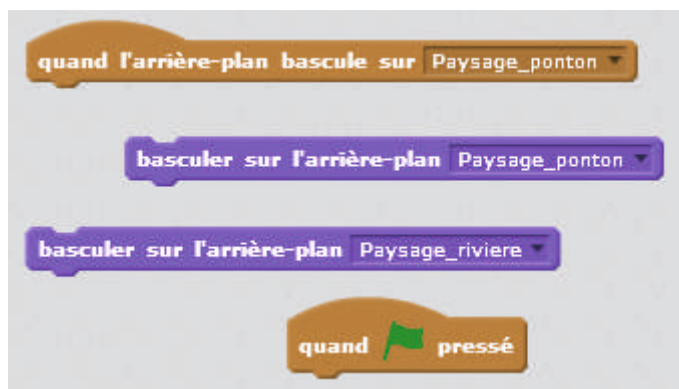
Résumé	Les élèves font en sorte que les deux premiers épisodes de l'aventure du héros s'enchaînent. Pour cela, ils découvrent la notion de message : un message peut être diffusé lors de l'exécution d'une instruction, et la réception de ce message peut déclencher une ou plusieurs instructions.
Notions (cf. scénario conceptuel, page 114)	« Algorithmes » <ul style="list-style-type: none">• Des instructions peuvent démarrer en même temps, si leur exécution est déclenchée par un même événement.
Matériel	Idem séances précédentes
Lexique	Sous-programme, événement
Durée	45 min

Situation déclenchante

L'enseignant rappelle que pour le moment, le programme du héros correspondant à l'épisode de la rivière est désactivé (aucun événement ne le déclenche). On voudrait pourtant que les deux épisodes du récit s'enchaînent. C'est le travail d'aujourd'hui.

Recherche (idéalement par binômes)

L'enseignant indique les instructions que les élèves vont devoir utiliser :



Il les laisse décider en toute autonomie de la façon de les intégrer à leur programme pour obtenir le résultat attendu.

Mise en commun

La mise en commun permet de discuter des difficultés rencontrées et des solutions trouvées par les élèves.

La possibilité la plus simple est la suivante.
Dans la zone de programmation de la scène on initialise la scène pour le début du récit :



À la toute fin du premier sous-programme du héros (qui marque la fin de l'épisode de la rivière), on ajoute une instruction « Basculer sur l'arrière-plan *Paysage_ponton* ».

On déclenche le sous-programme du héros pour l'épisode de la rivière par l'événement « Clic sur le drapeau vert ».

On déclenche les sous-programmes du héros et de la pieuvre pour l'épisode du ponton par l'événement de bascule de l'arrière-plan, en plaçant en tête de ces scripts l'instruction « Quand l'arrière-plan bascule sur *Paysage_ponton* ».

Il faut également compléter les initialisations, en particulier, lors du clic sur le drapeau vert, cacher la pieuvre et mettre la taille du héros à 100 %.

Le fichier « Il-2bis_ponton_correction_seance_5.sb2 » fournit un exemple de ce que l'on obtient pour l'ensemble des deux premiers épisodes (épisode de la rivière et épisode du ponton connectés).

Conclusion et traces écrites

La classe synthétise collectivement ce qui a été appris lors de cette séance.

- *Les sous-programmes sont déclenchés par des événements. Dans un même programme, un seul événement peut déclencher plusieurs sous-programmes à la fois.*

Les élèves notent ces conclusions dans leur cahier de sciences. Puis ils complètent leur fiche d'utilisation de *Scratch* en coloriant les instructions découvertes lors de cette séance et en écrivant des mots-clés : diffusion de message, événement « réception de message ».



Séance 6 – Différents types de boucles

Résumé	Les élèves racontent la suite de l'aventure du héros : la pieuvre descend chercher le trésor au fond de la mer et remonte avec. Ils renforcent les notions introduites aux séances précédentes, notamment celle de boucle prédéfinie, et découvrent la notion de boucle infinie.
Notions (cf. scénario conceptuel, page 105)	« Algorithmes » <ul style="list-style-type: none">• Certaines boucles sont répétées indéfiniment.• Certaines boucles sont répétées jusqu'à ce qu'une condition soit remplie.
Matériel	Idem séance précédente
Lexique	Boucle infinie
Durée	1 h + 30 min pour assurer la connexion entre les épisodes 2 et 3.

Situation déclenchante

L'enseignant rappelle aux élèves que le héros, incapable d'atteindre lui-même le trésor qu'il voit au fond de la mer, guide un sous-marin, représenté ici par une pieuvre. Celui-ci remonte avec le trésor. Aujourd'hui, les élèves vont programmer cet épisode de l'histoire du héros.

Recherche (idéalement en binômes)

L'enseignant donne aux élèves leur mission de programmation : dans le même programme que la dernière fois, ils doivent ajouter une troisième scène, qui apparaît une fois que l'hippocampe est visible dans l'océan. Sur cette scène, ils doivent placer un animal « décoratif » de leur choix, qui ne participe pas directement à l'histoire, et dont les mouvements se répètent pendant toute la scène. Il leur montre le résultat attendu (fichier « Il-2bis_mer.sb2 »), en signalant qu'il a utilisé de nouveaux types d'instructions de la catégorie « contrôle ».

Il discute avec la classe de la nature de ces instructions : ce sont des boucles, mais on ne décide pas à l'avance combien de fois elles vont s'exécuter. Certaines de ces boucles ont un espace en forme d'hexagone qu'il faut remplir avec une instruction de cette même forme. Il y en a dans la catégorie « capteurs » (et dans la catégorie « opérateurs », mais ce n'est pas utile ici), par exemple :



On obtient des combinaisons comme celle ci-contre, que l'enseignant peut interpréter avec la classe et tester sur le lutin chat.



Le lutin chat avance de 10, attend 1 seconde et recommence jusqu'à ce qu'il touche le bord. S'il touche le bord dès le départ, il ne fait donc rien. S'il démarre près du bord, il fera

peut-être un ou deux déplacements; s'il démarre loin du bord, il fera davantage de déplacements.

Cela étant compris, l'enseignant conseille de commencer par inactiver les sous-programmes des deux premiers épisodes du récit (pour cela, il suffit de les déconnecter de l'événement déclencheur), pour se concentrer sur l'épisode en cours... Il faut essayer d'obtenir un résultat semblable, mais pas nécessairement identique, à ce qui a été montré par l'enseignant.

Mise en commun

Lors de la mise en commun, l'enseignant revient sur les points qui ont pu poser problème. Il oriente la discussion sur les nouveaux types de boucles : la boucle « répéter indéfiniment » et la boucle « répéter jusqu'à... ». Si le temps le permet, il laisse aux élèves la possibilité d'améliorer leur première proposition à l'issue de la mise en commun. Sinon, les élèves pourront faire ces améliorations pendant la séance annexe de 30 minutes qui leur permettra aussi d'assurer la connexion entre les épisodes 2 et 3.

Le fichier « Il-2bis_mer_correction_seance_6.sb2 » donne un exemple de ce que l'on peut obtenir, une fois les sous-programmes des deux premiers épisodes réactivés et une fois la liaison entre épisodes du ponton et de la mer programmée.

Note pédagogique

Certains élèves vont utiliser des instructions que la classe n'a pas encore rencontrées. À ce stade où ils commencent à bien maîtriser le logiciel, il est important de laisser leur créativité s'exprimer. Mais pour faciliter leur découverte, l'enseignant doit leur conseiller de tester régulièrement leurs programmes : il est en effet beaucoup plus difficile de détecter un bug dans un programme que l'on a écrit d'un bloc sans tests que dans un programme qui a été testé au fur et à mesure de son écriture.

Conclusion et traces écrites

La classe synthétise collectivement ce qui a été appris lors de cette séance :

- Certaines boucles sont répétées indéfiniment.
- Certaines boucles sont répétées jusqu'à ce qu'une condition soit vérifiée (par exemple, contact d'un lutin avec un autre lutin, contact d'un lutin avec une certaine couleur).

Les élèves notent cette conclusion dans leur cahier de sciences et complètent la fiche d'utilisation de Scratch en coloriant les instructions nouvellement rencontrées et en notant quelques mots-clés : boucle infinie, condition, contact avec un lutin, contact avec une couleur.



Séance 7 – Produire l'épisode final en autonomie

Résumé	Les élèves sont en autonomie pour raconter un dernier épisode de l'aventure de leur héros. Ils réinvestissent les notions rencontrées au cours de la séquence et achèvent leur production.
Notions (cf. scénario conceptuel, page 114)	Idem séances précédentes
Matériel	Idem séances précédentes
Lexique	
Durée	Une ou plusieurs séances de 1 h.

Situation déclenchante

L'enseignant annonce aux élèves qu'ils ont presque toute liberté pour raconter comme ils le souhaitent un dernier épisode de l'aventure, avec *Scratch*. Une seule contrainte : ils vont devoir utiliser des événements pour coordonner les actions de leurs personnages et employer des boucles. L'enseignant conseille aux élèves de ne pas être trop ambitieux au départ et d'enrichir leur épisode dans un second temps s'ils en ont le temps.

Conception de l'épisode (idéalement en binômes)

Les élèves discutent, au sein de leur binôme, des éléments essentiels qu'ils souhaitent voir apparaître dans leur épisode. Quels personnages ? Quelle(s) scène(s) ? Quels événements déclencheurs des actions des personnages ?

Les élèves ont deux arrière-plans à disposition : la plage au coucher de soleil et la chambre du héros. Mais ils peuvent choisir autre chose dans les bibliothèques de *Scratch*.

Programmation de l'épisode

Après accord de l'enseignant, qui vérifie avec les élèves que leur projet est réaliste, les élèves programment effectivement leur épisode, dans un fichier indépendant de celui des trois premiers épisodes. L'enseignant tourne de groupe en groupe pour s'assurer que chacun progresse bien.

Conclusion : présentation aux autres élèves

En fin de travail, les élèves visionnent sur grand écran le récit du dernier épisode proposé par les différents groupes. Les élèves expliquent les programmes dont ils sont les plus fiers, ou pour lesquels ils ont eu des difficultés qu'ils ont su résoudre. Ils peuvent également partager des difficultés qui persistent et voir si la discussion permet de trouver des solutions. Un temps est donc prévu pour que les élèves puissent modifier leur épisode après la phase d'échange.

Le fichier « Il-2bis_magicien_correction_seance_7.sb2 » donne un exemple de programme permettant de raconter le dernier épisode (oiseau, magicien et retour à la maison du héros). Ce n'est qu'une possibilité parmi bien d'autres.

Prolongement

Maintenant que les élèves ont appris à programmer à l'aide de *Scratch*, ils peuvent réinvestir ces compétences pour d'autres productions libres à réaliser en cours d'année: une carte animée pour Noël, la fête des Mères, etc.

Cette ressource est issue du projet thématique **1,2,3... CODEZ !**, paru aux Éditions Le Pommier.

Claire Calmet, Mathieu Hirtzig et David Wilgenbus

1,2,3... CODEZ !

Enseigner l'informatique à l'école et au collège
(cycles 1, 2 et 3)

FONDATION La main à la pâte
POUR L'ÉDUCATION À LA SCIENCE

Qu'il s'agisse de préparer les enfants aux métiers de demain, de les aider à comprendre le monde numérique dans lequel ils vivent – afin qu'ils soient en mesure d'agir sur lui et non de le subir –, de les sensibiliser aux enjeux de citoyenneté, ou encore de favoriser la coopération ou développer leur créativité... l'informatique doit être enseignée à tous, dès le plus jeune âge.

Le projet « 1, 2, 3... codez ! » développé par la Fondation *La main à la pâte*, Inria et France 101 vise à initier les élèves et leurs enseignants à la science informatique, de la maternelle à la classe de 6^e. Il propose à la fois des activités branchées (nécessitant un ordinateur, une tablette ou un robot) permettant d'introduire les bases de la programmation et des activités débranchées (informatique sans ordinateur) permettant d'aborder des concepts de base de la science informatique (algorithme, langage, information...).

Un outil clés en main
Ce guide pédagogique comporte :

- 3 progressions pour la classe (cycles 1, 2 et 3)
 - Une approche pluridisciplinaire associant démarche d'investigation et pédagogie de projet ;
 - Des séances clés en main, testées en classe, organisées en séquences thématiques et scénarisées pour chaque cycle ;
 - Des fiches documentaires à photocopier ;
- Des éclairages pédagogiques et scientifiques pour guider l'enseignant dans la mise en œuvre du projet.

Les auteurs
Claire Calmet est formatrice et responsable des liens avec le monde de l'entreprise et de la recherche à la Fondation *La main à la pâte*.
Mathieu Hirtzig est webmestre et médiateur scientifique à la Fondation *La main à la pâte*.
David Wilgenbus est formateur et responsable de la production de ressources à la fondation *La main à la pâte*. Il coordonne le projet « 1, 2, 3... codez ! ».

FONDATION La main à la pâte
Lancée en 1996 par Georges Charpak, prix Nobel de physique, avec le soutien de l'Académie des sciences et du ministère de l'Éducation nationale, *La main à la pâte* vise à promouvoir à l'école primaire un enseignement de science et de technologie de qualité. <http://www.fondation-lamap.org>

Avec le soutien de :

Illustration : Catherine Zimmmermann

Éditions Le Pommier

74651106
21 €
Diffusion Bélin

Retrouvez l'intégralité de ce projet sur : <https://www.fondation-lamap.org/projets-thematiques>.

Fondation *La main à la pâte*

43 rue de Rennes
75006 Paris
01 85 08 71 79
contact@fondation-lamap.org

Site : www.fondation-lamap.org

FONDATION
La main à la pâte
POUR L'ÉDUCATION À LA SCIENCE