

# Quelques défis pour débuter avec Scratch

Vous trouverez à la suite **une présentation de l'interface du logiciel Scratch3** (disponible à cette adresse <https://scratch.mit.edu> ou téléchargeable en [cliquant ici](#)) ainsi que **des défis de difficulté croissante** qui vous permettront de débuter progressivement avec vos élèves et de leur faire découvrir les principales fonctions de *Scratch*. A chaque mission des niveaux 1 et 2 sont associées des petites cartes d'aide (à découper pour pouvoir les distribuer séparément) que je vous conseille de ne donner aux élèves qu'en cas de « blocage » total car elles présentent tous les blocs à utiliser pour réussir la mission (et donc quasiment la solution au défi). Pour les missions du niveau 3 les fiches d'aide associées permettent de guider les élèves étape par étape afin de leur permettre de trouver une solution (à ne donner également uniquement qu'en cas de « blocage »).

**L'idée est de laisser les élèves tâtonner et faire des essais afin de trouver une solution** (il peut y en avoir plusieurs) **qui réponde aux missions proposées.**

Ce document est voué à être complété à l'avenir (par moi-même ou par certains d'entre vous ou de vos élèves). N'hésitez pas à me contacter pour me faire parvenir vos remarques ou idées qui, j'en suis persuadé seront très constructives et pertinentes.

Comme l'indique le logo en bas de chaque page, ce document est sous licence *Creative Commons* qui vous autorise à le reproduire, à le partager et à le modifier à conditions que ce ne soit pas à des fins commerciales, que le partage (ou la diffusion) soit effectué sous la même licence que cet original et que l'auteur original soit cité.

Ces défis permettent de travailler les compétences suivantes :

CRCN (Cadre de Référence des Compétences Numériques) :



**3.4. Programmer :** Écrire des programmes et des algorithmes pour répondre à un besoin (automatiser une tâche répétitive, accomplir des tâches complexes ou chronophages, résoudre un problème logique...) et pour développer un contenu riche (jeu, site web...) (avec des environnements de développement informatique simples, des logiciels de planification de tâches...).

Compétences transversales :

**Domaine 1 : les langages pour penser et communiquer**

Comprendre, s'exprimer en utilisant les langages mathématiques, scientifiques et informatiques  
Manipuler le clavier pour taper du texte

**Domaine 2 : Les méthodes et les outils pour apprendre**

Utiliser l'ordinateur

Coopérer et réaliser des projets

**Domaine 5 : Les représentations du monde et l'activité humaine**

Programmer les déplacements d'un personnage sur un écran

<u>Sommaire</u> :	Eléments de programmation utilisés dans les défis .....	p3
	L'interface .....	p4
	Conseils .....	p7
	Défis niveau 1 :	
	Fiches missions .....	p8
	<i>Mission n°1 : Déplacement / Repérage</i>	
	<i>Mission n°2 : Communication / Apparence</i>	
	Cartes AIDE mission n°1 .....	p9
	Cartes AIDE mission n°2.....	p10
	Défis niveau 2 :	
	Fiche mission n°1 .....	p11
	<i>Apparence / Déplacement / Repérage / Boucle / Programmation de touches</i>	
	Fiche mission n°2.....	p12
	<i>Apparence / Son / Boucle / Déplacement / Repérage / Programmation de touches</i>	
	Fiche mission n°3.....	p13
	<i>Apparence / Condition / Déplacement / Boucle / Programmation de touches / Communication / Temporisation</i>	
	Cartes AIDE mission n°1 .....	p14
	Cartes AIDE mission n°2.....	p15
	Cartes AIDE mission n°3.....	p16
	Solutions possibles .....	p17
	Défis niveau 3 :	
	Fiche mission n°1 .....	p18
	<i>Apparence / Déplacement / Repérage / Boucle / (Condition)</i>	
	Fiche mission n°2.....	p18
	<i>Apparence / Boîte de dialogue / Condition / Test logique / Temporisation</i>	
	Fiche mission n°3.....	p19
	<i>Apparence / Repérage / Déplacement / Boucle / Programmation de touches / Variable / Condition / Nombre aléatoire</i>	
	Fiches AIDE mission n°1 .....	p20
	Fiches AIDE mission n°2 et 3.....	p21
	Solutions possibles .....	p22-23
	Utiliser les capteurs :	
	Fiche AIDE.....	p24
	Solution possible.....	p25

## *Eléments de programmation utilisés dans les défis*

### Déplacement :

Utilisation de blocs permettant le déplacement et l'orientation d'un sprite.

### Repérage :

Utilisation des coordonnées (x ; y) dans le repère orthonormé.

### Communication :

Un sprite communique un message écrit (bulle) ou oral (enregistrement d'un son).

### Apparence :

Action sur les costumes d'un sprite, sur les arrière-plans ou utilisation des effets graphiques.

### Boucle :

Utilisation d'au moins une boucle (répéter ...).

### Son :

Utilisation de sons dans le script.

### Programmation de touches :

Certaines touches du clavier sont programmées pour déclencher des actions.

### Condition :

Utilisation de capteurs permettant de vérifier si une condition est remplie ou pas.

### Temporisation :

Gestion des lancements des actions pour qu'elles démarrent à un moment précis ou durent un temps défini.

### Boîte de dialogue :

Utilisation d'une boîte de dialogue dans laquelle l'utilisateur peut entrer des données (une « réponse »).

### Test logique :

Utilisation des blocs « opérateurs » permettant de comparer deux éléments (texte ou numérique).

### Nombre aléatoire :

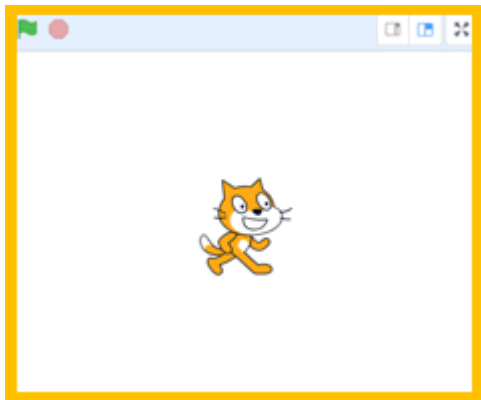
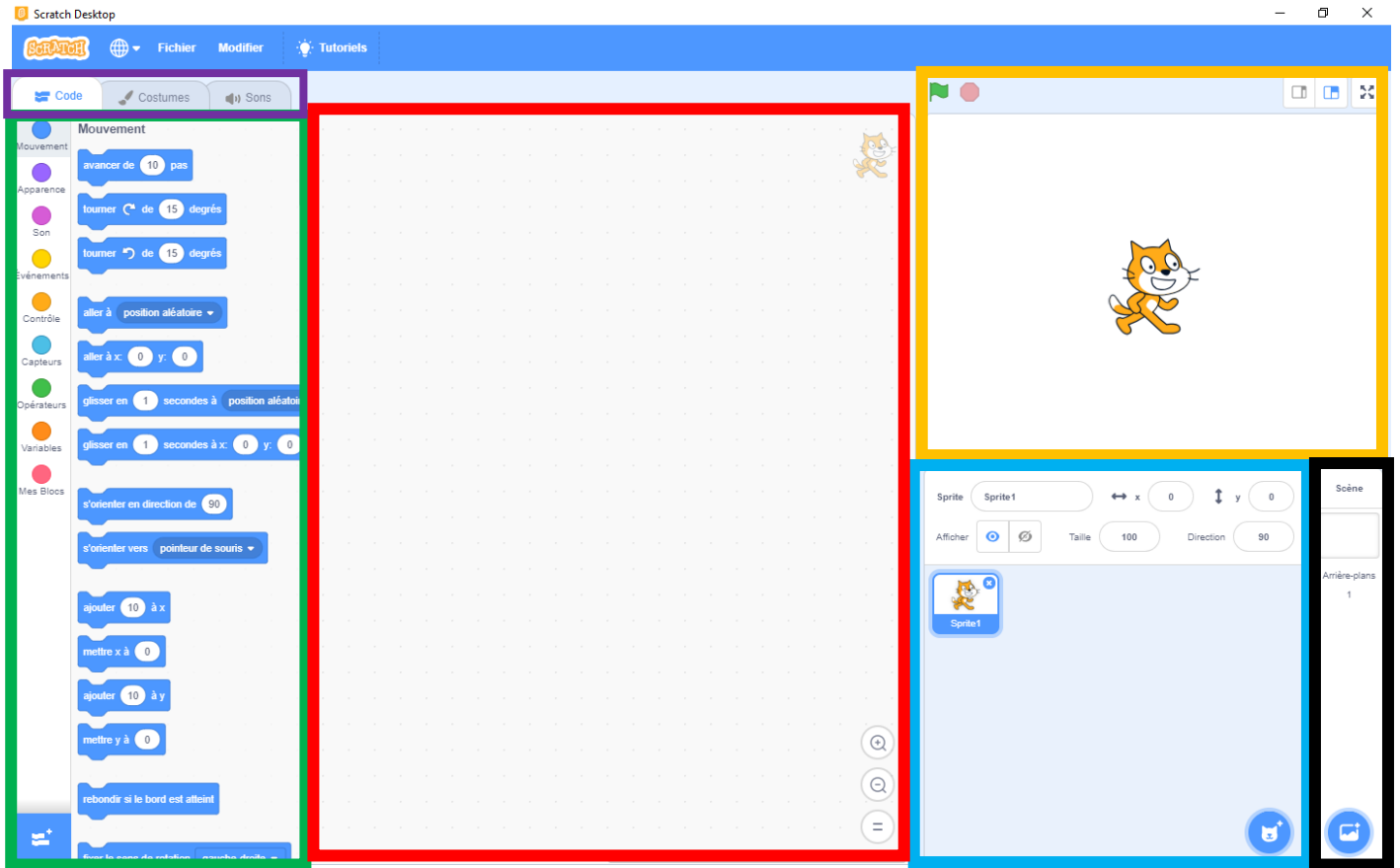
Utilisation de nombres aléatoires

### Variable :

Utilisation d'une variable (valeur que l'on fait varier au cours du programme ; par exemple un score ou un nombre choisi aléatoirement à un moment du programme).

		Déplacement	Repérage	Communication	Apparence	Boucle	Son	Programmation de touches	Condition	Temporisation	Boîte de dialogue	Test logique	Nombre aléatoire	Variable
Niveau 1	Mission 1													
	Mission 2													
Niveau 2	Mission 1													
	Mission 2													
	Mission 3													
Niveau 3	Mission 1													
	Mission 2													
	Mission 3													

# L'interface

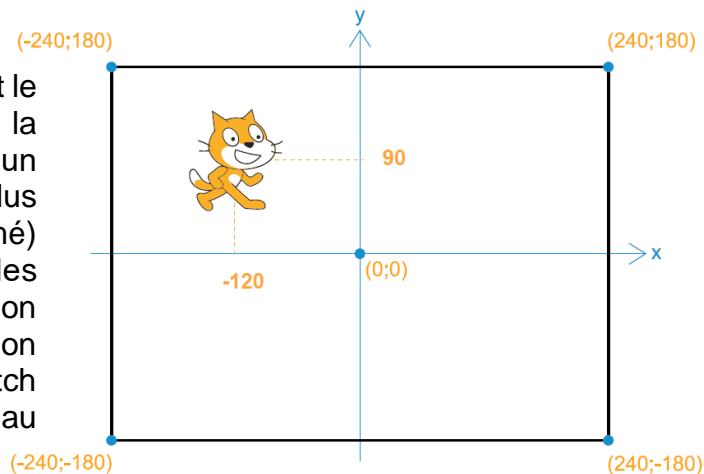


## La scène :

C'est la zone où l'on peut voir le résultat de la programmation. On peut y afficher des arrière-plans, des objets ou des personnages à programmer.

## Se repérer sur la scène :

Pour faciliter le positionnement et le déplacement des sprites sur la scène, celle-ci est munie d'un quadrillage invisible (plus précisément un repère orthonormé) pour lequel chaque nœud a des coordonnées (x pour la position horizontale et y pour la position verticale). Au démarrage Scratch est situé au centre de la scène, au point (x=0 ; y=0).



## Taille de la scène


480 x 360 pixels

## Repère orthonormé

Unité = 1 pixel  
Centre = centre de la scène




### La zone des sprites :

Les sprites sont les objets et les personnages que l'on peut programmer. Lorsqu'on lance le logiciel, le Sprite1 (le chat *Scratch*) est automatiquement ajouté. Pour en ajouter d'autres, il suffit de passer la souris sur cette icône  pour ouvrir le menu et choisir son sprite.



### La zone des arrière-plans :

Dans cette zone s'affiche l'arrière-plan actif (visible dans la scène) et en dessous le nombre d'arrière-plans qui ont été « chargé ». Au lancement du logiciel, c'est *l'arrière plan 1* (tout blanc) qui est actif.

Pour charger d'autres arrière-plans, il suffit de passer la souris sur cette icône  pour ouvrir le menu et choisir un arrière-plan.



### La colonne des commandes :

C'est ici que se trouvent tous les blocs de programmation. Chaque bloc correspond à un ordre (une commande). Les blocs sont classés dans 9 catégories qui ont chacune une couleur spécifique. Pour essayer un bloc, il suffit de cliquer une fois dessus.

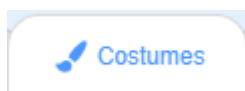
## La zone de codage :

C'est ici que l'on écrit les programmes en faisant glisser des blocs de la colonne des commandes et en les assemblant. Pour commencer un programme, il faudra toujours un évènement déclencheur (dans la catégorie Evénements). L'arrière-plan ou le sprite qui est programmé est représenté en filigrane.

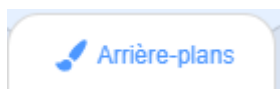


## Les onglets :

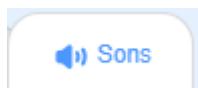
Ils permettent de passer de la fenêtre de code (avec la colonne des commandes et la zone de codage) aux fenêtres de costumes, d'arrière-plan et de son.



L'onglet Costumes permet de voir les différents « costumes » (images correspondantes au sprite sélectionné), de les modifier ou d'en créer de nouveaux.



L'onglet Arrière-plans permet de voir les différents arrière-plans ayant été « chargés », de les modifier et d'en créer de nouveaux.



L'onglet Sons permet d'ajouter des sons (pour qu'ils apparaissent dans la liste déroulante des blocs « jouer le son ... ») issus de l'application ou de votre ordinateur. Si vous êtes équipé d'un micro (les ordinateurs portables en ont souvent un intégré), vous pouvez aussi vous enregistrer et utiliser ce son dans votre programme. Dans cette fenêtre, vous trouverez également des outils vous permettant de modifier les sons (en les accélérant, les ralentissant, ou en appliquant d'autres effets).

## Les menus :



Regarder des tutoriels vidéos (en anglais sous-titrés en français)

Modifier la langue de l'interface

- Faire réapparaître le dernier sprite ou arrière-plan supprimé
- Activer le mode Turbo

- Créer un nouveau projet
- Ouvrir un projet enregistré sur l'ordinateur
- Enregistrer le projet en cours

## Conseils :

- ❖ **Si vos élèves n'ont jamais utilisé Scratch**, il peut être intéressant (voire nécessaire) de **faire une séance préalable de découverte de l'interface**. Cette séance peut être menée en classe entière si vous disposez d'un TBI (ou simplement d'un vidéoprojecteur). Laissez les élèves observer, puis demandez-leur de décrire ce qu'ils voient. Profitez-en pour leur **apporter le vocabulaire précis** pour faciliter les futurs échanges (il est plus simple de trouver le drapeau vert si on dit qu'il est « en haut à gauche de la scène » plutôt que de dire qu'il est « presque au milieu en haut »...).
- ❖ Par expérience, **l'organisation la plus efficace** pour faire réaliser ces défis aux élèves est de les mettre par **groupe de 2** (en veillant à ce qu'il n'y ait pas de trop grande différence de niveau entre ces 2 élèves). Il peut parfois être nécessaire d'instaurer des règles pour établir un « tour de manipulation de l'ordinateur » en expliquant l'importance du rôle de l'élève qui ne manipule pas pour vérifier ce qui est fait et apporter son aide. Par exemple, on peut proposer un changement à chaque étape d'un défi (un élève prend la souris pour l'étape a) et l'autre pour l'étape b).
- ❖ Comme le temps passe toujours vite quand on fait de la programmation, je vous conseille de prendre un petit temps pour **expliquer aux élèves comment enregistrer leur projet avant de commencer à programmer** (ce qui fera gagner du temps en fin de séance).
  - Avec le logiciel installé :  
Leur indiquer dans quel dossier enregistrer leur programme et le nom à lui donner. Par exemple, les prénoms des élèves créateurs, suivis du titre du projet. Une fois la séance terminée, ils auront juste à cliquer sur « Sauvegarder sur votre ordinateur », puis sur « Enregistrer » dans la fenêtre qui va s'ouvrir et enfin sur « Oui » pour confirmer le remplacement de la version précédente.
  - Avec la version en ligne : vous avez 2 possibilités pour enregistrer :
    1. Enregistrer les projets en ligne : créez un compte pour la classe au préalable, connectez-vous, puis remplacez « Untitled » par les prénoms des élèves créateurs, suivis du titre du projet. Cliquez sur « Fichier » puis « Enregistrer maintenant ». Vous retrouverez les projets enregistrés en ligne en cliquant sur le dossier un haut à droite (à côté de votre identifiant de connexion).
    2. Enregistrer les projets sur l'ordinateur (vous pourrez ensuite les rouvrir en cliquant sur « Importer depuis votre ordinateur ») : le projet sera enregistré dans votre dossier de téléchargement sous le nom « Projet Scratch ». Pensez à dire à vos élèves d'aller renommer le fichier en fin de séance (après leur dernier enregistrement) avec leurs prénoms, suivis du titre du projet, pour pouvoir le retrouver plus facilement lors des séances suivantes.
- ❖ **Lors de la première séance**, il vous faudra trouver une organisation de classe (fonctionnement en atelier par exemple) qui vous permettra d'être assez disponible pour être certain que les élèves qui réalisent les défis ne se dispersent pas, pour pouvoir leur donner de petits indices pour les faire avancer et surtout **les inciter à faire des essais**. Comme indiqué dans l'introduction de ce document ne donnez les fiches « Aide » qu'en dernier recours ou si vous ne pouvez pas vous rendre disponible pour aider les élèves « bloqués ». **Pour les séances suivantes, les élèves devraient pouvoir être autonomes**. Vous aurez juste à donner les fiches « Aide » après un délai raisonnable de recherche infructueuse.
- ❖ Si vous en avez la possibilité, imprimez les fiches « Aide » en couleurs (elles seront plus lisibles pour les élèves et leur permettront de trouver plus facilement les blocs à utiliser) et plastifiez-les afin de pouvoir les conserver dans le temps.

## Et après ?

Rendez-vous sur le site Numérique éducatif 80 (<http://tice.dsden80.ac-amiens.fr/>) pour retrouver les défis programmation proposés par l'équipe des ERUN de la Somme depuis 2017 ou découvrez les [Défis mathématiques avec Scratch](#).

Nom du groupe : \_\_\_\_\_

## NIVEAU 1

<b>Mission n°1a</b>	<b>Scratch se déplace de gauche à droite quand on clique sur le drapeau vert.</b> <i>Déplacement / Repérage</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi	<b>Mission n°2a</b>	<b>Scratch se présente quand on clique sur lui.</b> <i>Communication</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi
<b>Mission n°1b</b>	<b>Scratch se déplace de gauche à droite puis de bas en haut quand on clique sur le drapeau vert.</b> <i>Déplacement / Repérage</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi	<b>Mission n°2b</b>	<b>Scratch se présente et change de « costume » quand on clique sur lui.</b> <i>Apparence</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi
<b>Mission n°1c</b>	<b>Scratch se déplace de gauche à droite puis de bas en haut et fait demi-tour quand on clique sur le drapeau vert.</b> <i>Déplacement / Repérage</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi	<b>Mission n°2c</b>	<b>Scratch se présente, change de « costume » et rétrécit quand on clique sur lui.</b> <i>Apparence</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi



Nom du groupe : \_\_\_\_\_

## NIVEAU 1

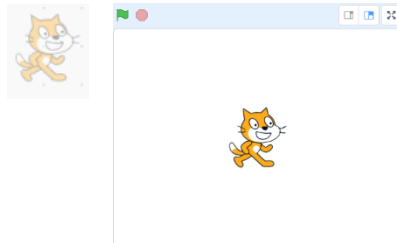
<b>Mission n°1a</b>	<b>Scratch se déplace de gauche à droite quand on clique sur le drapeau vert.</b> <i>Déplacement / Repérage</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi	<b>Mission n°2a</b>	<b>Scratch se présente quand on clique sur lui.</b> <i>Communication</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi
<b>Mission n°1b</b>	<b>Scratch se déplace de gauche à droite puis de bas en haut quand on clique sur le drapeau vert.</b> <i>Déplacement / Repérage</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi	<b>Mission n°2b</b>	<b>Scratch se présente et change de « costume » quand on clique sur lui.</b> <i>Apparence</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi
<b>Mission n°1c</b>	<b>Scratch se déplace de gauche à droite puis de bas en haut et fait demi-tour quand on clique sur le drapeau vert.</b> <i>Déplacement / Repérage</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi	<b>Mission n°2c</b>	<b>Scratch se présente, change de « costume » et rétrécit quand on clique sur lui.</b> <i>Apparence</i>	<input type="checkbox"/> Réussi seul <input type="checkbox"/> Avec aide <input type="checkbox"/> Non réussi



Niveau 1

### Mission n°1a

Scratch se déplace de gauche à droite quand on clique sur le drapeau vert.  
*Déplacement / Repérage*



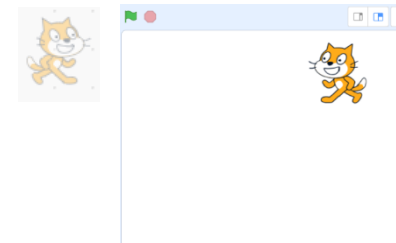
AIDE



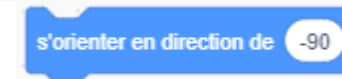
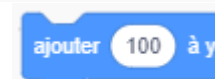
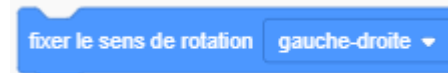
Niveau 1

### Mission n°1c

Scratch se déplace de gauche à droite puis de bas en haut et fait demi-tour quand on clique sur le drapeau vert.  
*Déplacement / Repérage*



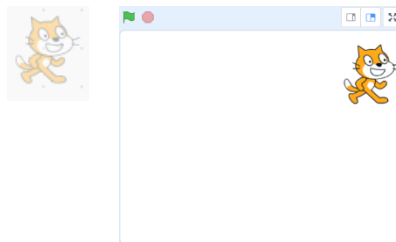
AIDE



Niveau 1

### Mission n°1b

Scratch se déplace de gauche à droite puis de bas en haut quand on clique sur le drapeau vert.  
*Déplacement / Repérage*



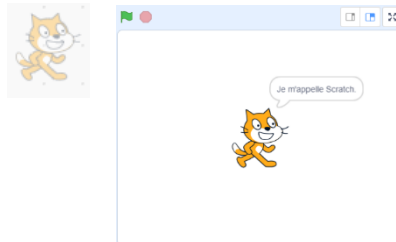
AIDE



Niveau 1

### Mission n°2a

Scratch se présente  
quand on clique sur lui.  
*Communication*



AIDE

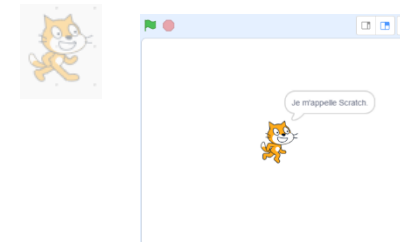
dire Je m'appelle Scratch.

quand ce sprite est cliqué

Niveau 1

### Mission n°2c

Scratch se présente,  
change de « costume » et  
rétrécit quand on clique  
sur lui.  
*Apparence*



AIDE

costume suivant

ajouter -10 à la taille

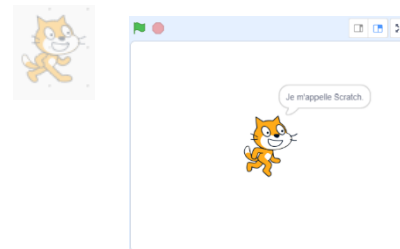
quand ce sprite est cliqué

dire Je m'appelle Scratch.

Niveau 1

### Mission n°2b

Scratch se présente et  
change de « costume »  
quand on clique sur lui.  
*Apparence*



AIDE

costume suivant



quand ce sprite est cliqué

dire Je m'appelle Scratch.

## NIVEAU 2

### Mission n°1

- a) **Insère l'arrière-plan Wall1.**  
*Apparence*
- b) **Scratch se place en  $x = -194$  et  $y = -74$  quand le drapeau vert est cliqué.**  
*Déplacement / Repérage*
- c) **Scratch se déplace du bord gauche de l'écran au bord droit de l'écran en courant sur le trottoir (ses jambes et ses bras doivent bouger pour donner l'impression de mouvement) quand la touche espace est pressée.**  
*Déplacement / Apparence / Boucle / Programmation de touches*
- d) **Quand on appuie sur la touche espace et que Scratch touche le bord de l'écran, il fait demi-tour et court dans l'autre sens.**  
*Déplacement / Apparence / Programmation de touches*

Auto-évaluation		Avec l'aide	
a) J'ai réussi à insérer l'arrière-plan « Wall1 ».			
b) J'ai réussi à programmer le déplacement de Scratch en $x=-194$ et $y=-74$ quand on clique sur le drapeau vert.			
c) J'ai réussi à programmer le déplacement de Scratch du bord gauche au bord droit de l'écran quand la touche espace est pressée.			
c) J'ai réussi à modifier le costume de Scratch pendant son déplacement pour donner l'impression de mouvement.			
c) J'ai utilisé une boucle (« répéter ... fois »).			
d) J'ai réussi à programmer le demi-tour de Scratch quand il touche le bord de l'écran.			
d) J'ai réussi à programmer le déplacement de Scratch de la droite vers la gauche après le demi-tour.			

Nom du groupe : \_\_\_\_\_

## NIVEAU 2

### Mission n°2

a) **Insère l'arrière-plan Spotlight.**

*Apparence*



b) **La musique « Reggae » joue en boucle quand le drapeau vert est cliqué.**

*Son / Boucle*

c) **Scratch se déplace quand on appuie sur les flèches de direction du clavier (vers le haut quand on appuie sur la touche « ↑ », à droite quand on appuie sur la touche « → », ...)**

*Déplacement / Repérage / Programmation de touches*

**Indice** : Programme les touches une par une.

Auto-évaluation		Avec l'aide	
a) J'ai réussi à insérer l'arrière-plan « Spotlight ».			
b) J'ai réussi à faire démarrer le son « Reggae » quand on clique sur le drapeau vert.			
b) J'ai réussi à faire tourner en boucle le son « Reggae » (il ne s'arrête pas).			
c) J'ai réussi à programmer le déplacement de Scratch vers le haut quand on appuie sur la touche « haut ».			
c) J'ai réussi à programmer le déplacement de Scratch vers le bas quand on appuie sur la touche « bas ».			
c) J'ai réussi à programmer le déplacement de Scratch vers la droite quand on appuie sur la touche « droite ».			
c) J'ai réussi à programmer le déplacement de Scratch vers la gauche quand on appuie sur la touche « gauche ».			

Nom du groupe : \_\_\_\_\_



## NIVEAU 2

### Mission n°3

a) **Insère l'arrière-plan Moon.**  
*Apparence*

b) **Scratch se déplace en courant vers le pointeur de la souris** (ses jambes et ses bras doivent bouger pour donner l'impression de mouvement) **quand la touche espace est pressée et ne s'arrête que lorsqu'il a réussi à toucher le pointeur de la souris.**  
*Condition / Déplacement / Apparence / Boucle / Programmation de touches*

c) **Quand Scratch touche le pointeur de la souris, il dit « Attrapé ! » pendant 5 secondes.**  
*Condition / Communication / Temporisation*

Auto-évaluation		Avec l'aide	
a) J'ai réussi à insérer l'arrière-plan « Moon ».			
b) J'ai réussi à programmer le déplacement de Scratch vers le pointeur de la souris quand la touche espace est pressée.			
b) J'ai réussi à modifier le costume de Scratch pendant son déplacement pour donner l'impression de mouvement.			
b) J'ai utilisé une boucle (« répéter jusqu'à ce que ... »).			
b) J'ai réussi à programmer l'arrêt de Scratch quand il touche le pointeur de la souris.			
c) J'ai réussi à programmer l'apparition d'une bulle « Attrapé ! » pendant 5 secondes quand Scratch touche le pointeur de la souris.			

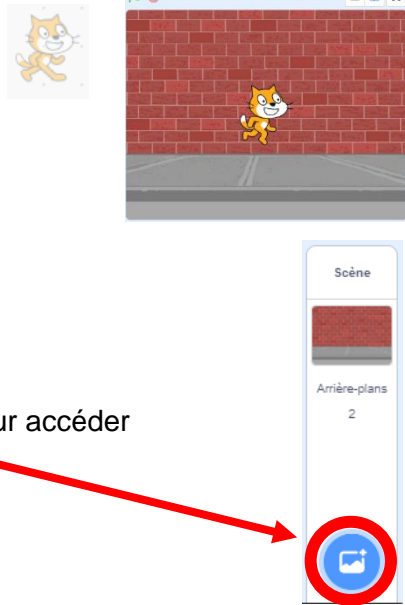
Niveau 2

### Mission n°1a

Insère l'arrière-plan Wall1.  
*Apparence*

AIDE

Clique deux fois (double-clic) ici pour accéder  
aux arrière-plans



Niveau 2

### Mission n°1c

Scratch se déplace du bord gauche de l'écran au bord droit de l'écran en courant sur le trottoir (ses jambes et ses bras doivent bouger pour donner l'impression de mouvement) **quand la touche espace est pressée.**  
*Déplacement / Apparence / Boucle / Programmation de touches*

AIDE



Niveau 2

### Mission n°1b

Scratch se place en  $x = -194$  et  $y = -74$  quand le drapeau vert est cliqué.  
*Déplacement / Repérage*

AIDE



Niveau 2

### Mission n°1d

Quand on appuie sur la touche espace et que Scratch touche le bord de l'écran, il fait demi-tour et court dans l'autre sens.  
*Déplacement / Apparence / Programmation des touches*

AIDE



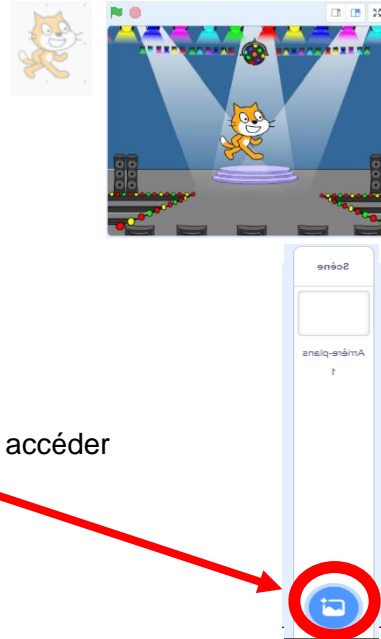
Niveau 2

Mission n°2a

Insère l'arrière-plan Spotlight.  
Apparence

AIDE

Clique deux fois (double-clic) ici pour accéder  
aux arrière-plans



Niveau 2

Mission n°2c

Scratch se déplace quand on appuie sur les  
flèches de direction du clavier (vers le haut  
quand on appuie sur la touche « ↑ », à droite  
quand on appuie sur la touche « → », ...)  
Déplacement / Repérage / Programmation de  
touches

AIDE

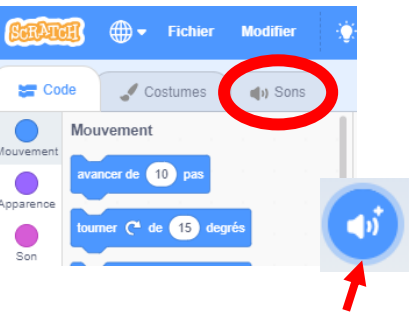


Niveau 2

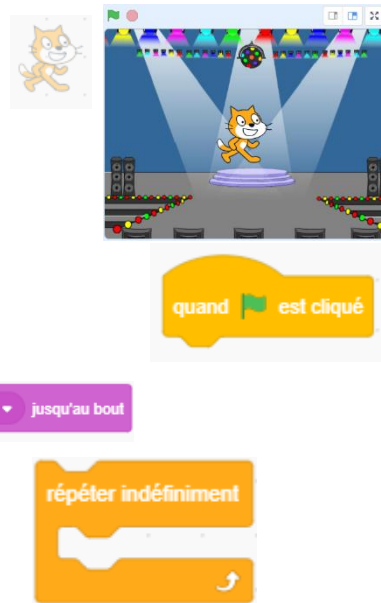
Mission n°2b

La musique « Reggae » joue en boucle  
quand le drapeau vert est cliqué.  
Son / Boucle

AIDE



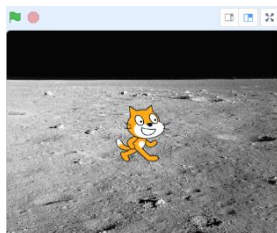
Clique deux fois (double-clic) ici  
pour accéder aux sons



Niveau 2

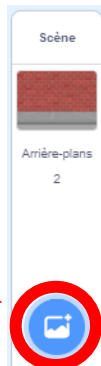
Mission n°3a

Insère l'arrière-plan Moon.  
Apparence



AIDE

Clique deux fois (double-clic) ici pour accéder  
aux arrière-plans

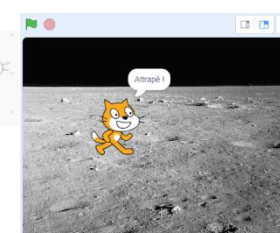


Niveau 2

Mission n°3c

Quand Scratch touche le pointeur de la  
souris, il dit « Attrapé ! » pendant 5  
secondes.

Condition / Communication / Temporisation



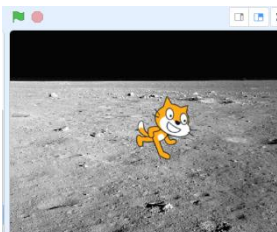
AIDE



Niveau 2

Mission n°3b

Scratch se déplace en courant vers  
le pointeur de la souris quand la  
touche espace est pressée et ne  
s'arrête que lorsqu'il a réussi à  
toucher le pointeur de la souris.  
Condition / Déplacement / Apparence /  
Boucle / Programmation de touches



AIDE





## NIVEAU 2

### Mission n°1 (solution possible)

```
when green flag clicked
  go to x: -194 y: -74

when space key is pressed
  set rotation direction to left-right
  repeat 39 times
    bounce if edge reached
    move 10 steps
    next costume
```

## NIVEAU 2

### Mission n°2 (solution possible)



```
when green flag clicked
  repeat indefinitely
    play sound Reggae until finished

when up arrow key is pressed
  add 10 to y

when left arrow key is pressed
  add -10 to x

when right arrow key is pressed
  add 10 to x

when down arrow key is pressed
  add -10 to y
```

## NIVEAU 2

### Mission n°3 (solution possible)

```
when space key is pressed
  repeat until mouse clicked
    move 10 steps
    turn towards mouse cursor
    next costume



say Attrapé! pendant 5 secondes
```

Nom du groupe : \_\_\_\_\_

### NIVEAU 3

#### Mission n°1

**Scratch traverse indéfiniment** (sans s'arrêter) **les 3 chambres** (arrière-plans *Bedroom 1*, *Bedroom 2* et *Bedroom 3*) quand le drapeau vert est cliqué.

Auto-évaluation		Avec l'aide	
a) J'ai réussi à afficher l'arrière-plan « Bedroom 1 ».			
b) J'ai réussi à programmer le déplacement de Scratch d'un côté à l'autre de la scène quand on clique sur le drapeau vert.			
b) J'ai réussi à programmer le changement d'arrière-plan (« Bedroom 2 » puis « Bedroom 3 ») quand Scratch touche le bord de la scène.			
b) J'ai réussi à programmer la répétition du déplacement de Scratch d'un côté à l'autre de la scène et le changement d'arrière-plan jusqu'à ce qu'on arrête le programme.			





Nom du groupe : \_\_\_\_\_

### NIVEAU 3

#### Mission n°2

**Sur l'arrière-plan Castle, le magicien (Wizard) pose une question** (de votre choix) **au personnage nommé Jaime** quand le drapeau vert est cliqué.

**Jaime répond à la question** (c'est l'utilisateur qui tape la réponse au clavier). **Si la réponse est correcte un cadeau (Gift) apparaît à la place du magicien (Wizard). Si la réponse est incorrecte, c'est Jaime qui disparaît.**

Auto-évaluation		Avec l'aide	
a) J'ai réussi à afficher l'arrière-plan « Castle ».			
a) J'ai réussi à afficher les personnages Wizard et Jaime sur la scène.			
b) J'ai réussi à programmer l'affichage de la question du magicien quand on clique sur le drapeau vert et l'ouverture d'une boîte de dialogue dans laquelle l'utilisateur peut entrer une réponse.			
c) J'ai réussi à programmer l'apparition du cadeau si la réponse est correcte.			
c) J'ai réussi à programmer la disparition du magicien si la réponse est correcte.			
c) J'ai réussi à programmer la disparition de Jaime si la réponse est incorrecte.			



Nom du groupe : \_\_\_\_\_

## NIVEAU 3

### Mission n°3

**Un requin** (qui regarde vers la gauche) **bouge indéfiniment verticalement** (de haut en bas puis de bas en haut) **au milieu de la scène sur l'arrière-plan *Underwater 2***. **Un poisson placé à gauche de l'écran est dirigé à l'aide des touches du clavier** (il ne peut pas reculer). **Un score est affiché**. Chaque fois que le poisson parvient à traverser la scène jusqu'au bord droit, un point est ajouté au score et le poisson réapparaît à une hauteur aléatoire à gauche de la scène. Si le poisson touche le requin, il disparaît et réapparaît à une hauteur aléatoire à gauche de la scène.

Conseil : Pense à réduire la taille du requin et du poisson pour que le jeu soit intéressant.

Auto-évaluation		Avec l'aide	
a) J'ai réussi à afficher l'arrière-plan « Underwater 2 ».			
a) J'ai réussi à placer les sprites Fish (poisson) et Shark (requin) sur la scène et à modifier leur taille.			
b) J'ai réussi à programmer le déplacement automatique et continu du requin de haut en bas puis de bas en haut au milieu de la scène quand on clique sur le drapeau vert.			
c) J'ai réussi à programmer le déplacement du poisson à l'aide des flèches du clavier (haut, bas et droite).			
d) J'ai réussi à afficher un compteur « score »			
e) J'ai réussi à programmer l'ajout d'un point au score chaque fois que le poisson touche le bord droit de l'écran.			
e) J'ai réussi à faire disparaître le poisson quand il touche le bord droit de la scène et à le faire réapparaître à une hauteur aléatoire à gauche de la scène.			
d) J'ai réussi à faire disparaître le poisson quand il touche le requin et à le faire réapparaître à une hauteur aléatoire à gauche de la scène.			

## Mission n°1

AIDE

Scratch glisse :

- a) **Sélectionne les arrière-plans *Bedroom 1*, *Bedroom 2* et *Bedroom 3* et supprime l'arrière-plan *arrière plan 1*.**

*Apparence*

- b) **Quand le drapeau vert est cliqué Scratch répète indéfiniment :**

- **Glisser du bord gauche de la scène au bord droit (le bord gauche est à  $x = -240$  et le droit à  $x = 240$ )**
- **Passer à l'arrière-plan suivant**
- **Aller au bord gauche de la scène ( $x = -240$ )**

*Déplacement / Repérage / Apparence / Boucle*

## Mission n°1

AIDE

Scratch court :

- a) **Sélectionne les arrière-plans *Bedroom 1*, *Bedroom 2* et *Bedroom 3* et supprime l'arrière-plan *arrière plan 1*.**

*Apparence*

- b) **Quand le drapeau vert est cliqué Scratch se place en  $x = -200$  et  $y = -80$ .**

**Puis Scratch répète indéfiniment :**

- **Avance de 10 pas.**
- **Change de costume**
- **Si Scratch touche le bord alors on passe à l'arrière-plan suivant et Scratch retourne en  $x = -200$  et  $y = -80$ .**

*Repérage / Déplacement / Apparence / Condition / Boucle*

### Niveau 3

## Mission n°2



### AIDE

- a) Sélectionne l'arrière-plan Castle2 et les 3 sprites : Jaime, Wizard (magicien) et Gift (cadeau). Place les sprites sur la scène (le cadeau doit être au même endroit que le magicien et être caché au début programme) et Jaime et le magicien doivent se regarder.

*Apparence*

- b) Le magicien pose une question (celle que tu veux) quand le drapeau vert est cliqué (utilise un bloc de la catégorie **Capteurs**)

*Boîte de dialogue*

- c) Si la réponse est correcte (utilise ces blocs :  et écrit la « bonne réponse » dans la partie de droite) alors le magicien disparaît et le cadeau apparaît. Sinon Jaime disparaît (le magicien peut envoyer un message à Jaime  pour qu'il disparaisse quand

il reçoit le message).

*Condition / Test logique / Apparence*

### Niveau 3

## Mission n°3

### AIDE

- a) Sélectionne l'arrière-plan Underwater 2 et les sprites : Fish (poisson) et Shark (requin). Mets le poisson à la taille 30, le requin à la taille 60 et oriente-le qu'il regarde vers la gauche. Place le poisson à gauche de la scène et le requin en  $x=0$ .

*Apparence / Repérage*

- b) Le requin se déplace de haut en bas puis de bas en haut indéfiniment quand on clique sur le drapeau vert.

*Déplacement / Repérage / Boucle*

- c) Le poisson se déplace grâce aux touches du clavier.

*Déplacement / Programmation de touches*

- d) Crée une variable nommée « Score » et affiche-la sur la scène.

*Variable*

- e) A chaque fois que le poisson touche le bord droit de la scène, un point est ajouté au score et le poisson réapparaît à gauche de la scène.

*Boucle / Condition / Variable / Déplacement / Repérage / Nombre aléatoire*

- f) A chaque fois que le poisson touche le requin, il réapparaît à gauche de la scène.

*Boucle / Condition / Déplacement / Repérage / Nombre aléatoire*

## NIVEAU 3

### Mission n°1

(solutions possibles)

Scratch glisse :



Scratch court :

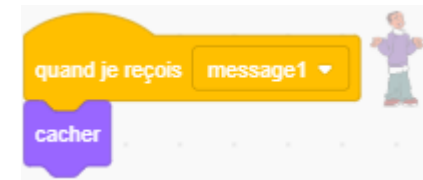


Nicolas VASSEUR  
ERUN DOULLENS

## NIVEAU 3

### Mission n°2

(solution possible)



## NIVEAU 3

### Mission n°3

(solution possible)

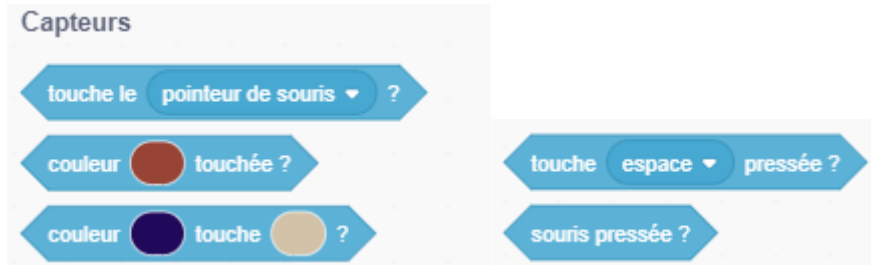
The script is composed of several event-driven blocks:

- when green flag is clicked:**
  - set Score to 0
  - go to x: -220 y: -11
  - repeat indefinitely:
    - if touches Shark?:
      - go to x: -240 y: random number between -180 and 180
- when up arrow is pressed:**
  - add 10 to y
- when right arrow is pressed:**
  - add 10 to x
- when down arrow is pressed:**
  - add -10 to y
- when green flag is clicked:**
  - repeat indefinitely:
    - if touches bord?:
      - add 1 to Score
      - go to x: -220 y: random number between -180 and 180

The script is composed of several event-driven blocks:

- when green flag is clicked:**
  - repeat indefinitely:
    - slide in 1 seconds to x: 0 y: -180
    - slide in 1 seconds to x: 0 y: 180

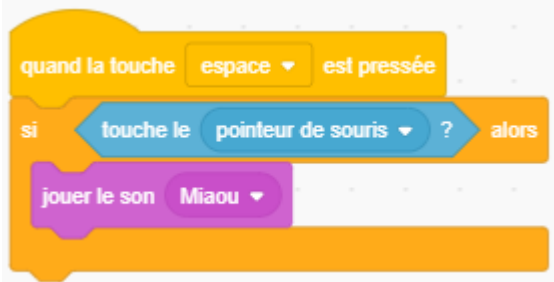
L'utilisation de capteurs permet de savoir si une couleur, un sprite, le pointeur de souris, le bord de la scène, sont touchés ou si une touche du clavier ou de la souris est pressée.




Ces blocs sont « en pointe » et s'insèrent dans les blocs de contrôle suivants :

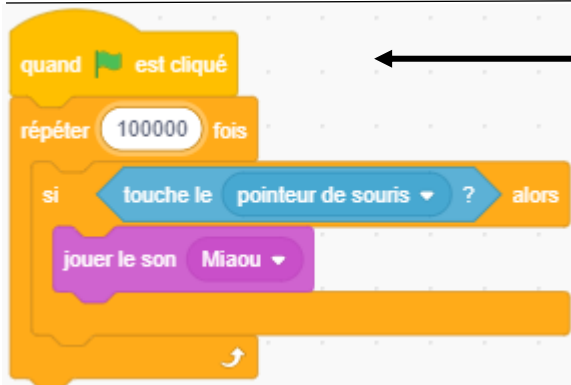


Crée le script suivant pour Scratch :



Place le pointeur de souris (la flèche) sur Scratch puis appuie sur la touche « espace ». **Que se passe-t-il ?**

Remplace le bloc « quand la touche espace est pressée » par . Clique sur le drapeau vert puis place le pointeur de souris sur Scratch. **Que se passe-t-il ?**



Programme ce script (on ajoute « répéter 100 000 fois »). Clique sur le drapeau vert puis place tout de suite le pointeur de souris sur Scratch. **Que se passe-t-il ?**

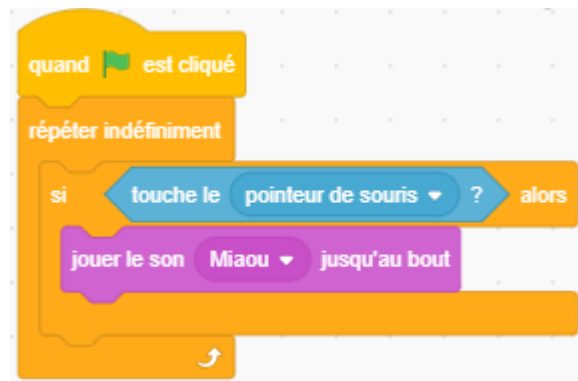
### Défi :

Une fois le drapeau vert cliqué, à chaque passage du pointeur de la souris sur Scratch, on entend le son « Miaou ».



## Utiliser les capteurs

(solution possible)



Remarque : Lorsqu'on clique sur le drapeau vert, les différents blocs situés sous l'évènement **quand le drapeau vert est cliqué** s'enchainent très rapidement. Ainsi, si on ne crée pas de boucle (« répéter indéfiniment »), le programme vérifie immédiatement après que le drapeau soit cliqué si le sprite touche le pointeur de souris. Si c'est le cas (ce qui est impossible car on ne peut pas bouger la souris assez vite pour être sur le sprite immédiatement après avoir cliqué sur le drapeau vert) le son « Miaou » est joué. Sinon le programme continue (comme il n'y a plus de bloc après dans notre cas, il s'arrête).