

# La programmation à l'école.



Circonscription  
de Beauvais Sud 60

Qu'est-ce que  
la programmation et  
pourquoi l'enseigner  
à l'école ?

Cadre de référence

Quelles activités pour  
le cycle 3 ?

Activités avec des  
robots

**La programmation**

Sitographie

# Pourquoi enseigner la programmation à l'école ?

- Tout enseignement doit permettre aux enfants, adolescents (et adultes !) de comprendre le monde dans lequel ils vivent et de les préparer à y jouer un rôle actif. Les récentes mutations de nos modes de communication, loisirs, interactions sociales, outils de production... sont intimement liées aux progrès de l'informatique, à tel point que l'on parle aujourd'hui d'un « monde numérique ». Ce nouveau monde porte en lui de nombreux espoirs de progrès technologique, de création d'emploi... mais aussi de craintes, en particulier dans le champ de l'éthique et du respect de la vie privée.
- Qu'il s'agisse de préparer les enfants aux métiers de demain, de les aider à comprendre les objets et réseaux qui les entourent - **afin de ne pas les subir passivement mais d'être en mesure d'agir sur eux** - 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.

# Quelques définitions

- **Un programme** : c'est ce qui permet de doter un ordinateur d'une certaine fonctionnalité, de lui faire exécuter des actions.
- Il faut alors pouvoir « communiquer » avec l'ordinateur dans le but de le programmer. Malheureusement, le seul langage qu'un ordinateur comprenne est le langage binaire, une suite de 0 et de 1. Et écrire des ordres à son ordinateur en utilisant de longues séries de 0 et de 1 s'avère vite fastidieux... On peut alors utiliser un **langage de programmation** (le C, le Java, le HTML et, dans le monde de l'éducation, le LOGO,...), sorte d'interprète compréhensible à la fois par nous, êtres humains, et par l'ordinateur.

## Coder et programmer ?

La différence entre « coder » et « programmer » est d'ordre lexical.

Pour les spécialistes, « coder » désigne le fait de mettre en place un programme dans un langage de programmation précis, tandis que « programmer » renvoie à l'activité de conception, en amont du code. Dans le vocabulaire courant, « coder » est synonyme de « programmer ».

# Qu'est-ce qu'un algorithme ?

- **Un algorithme**, c'est une séquence d'actions réalisant un traitement. L'algorithmique est ainsi la discipline qui consiste à analyser des problèmes, à les décomposer pour les résoudre.
- Par exemple, si le problème posé est « comment sortir une voiture du garage ? », un algorithme peut être :
  - 1. ouvrir la porte du garage
  - 2. prendre la clef de la voiture
  - 3. ouvrir la porte avant-gauche de la voiture
  - 4. entrer dans la voiture
  - 5. vérifier que la voiture est au point mort
  - 6. mettre le contact



Le cadre de référence

# Le Cadre de référence des compétences numériques

5 domaines

- Information et Données
- Communication et Collaboration
- Création de contenu
- Protection et sécurité
- Environnement numérique

**»» Cadre de référence des compétences numériques**

3.4 Programmer	<p><b>De quoi s'agit-il ?</b> 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, etc.) et pour développer un contenu riche (jeu, site web, etc.) avec des environnements de développement informatique simples, des logiciels de planification de tâches, etc...</p> <p><b>Référence au socle commun :</b> - Savoir que des langages informatiques sont utilisés pour programmer des outils numériques et réaliser des traitements automatiques de données (Domaine 1) ; - Connaître les principes de base de l'algorithmique et de la conception des programmes informatiques (Domaine 1) ; - Mettre en œuvre les principes de base de l'algorithmique pour créer des applications simples. (Domaine 1)</p> <p><b>Thématiques associées :</b> Algorithmie et programme; représentation et codage de l'information; complexité; pensée algorithmique et informatique; collecte et exploitation de données massives; intelligence artificielle et robots.</p>			
	Niveau 1	Niveau 2	Niveau 3	Niveau 4 <i>(en cours)</i>
	Lire et construire un algorithme qui comprend des instructions simples.	Réaliser un programme simple.	Développer un programme pour répondre à un problème à partir d'instructions simples d'un langage de programmation ;  Modifier un algorithme simple en faisant évoluer ses éléments de programmation ;  Mettre au point et exécuter un programme simple commandant un système réel ou un système numérique.	Écrire un algorithme et développer un programme complexe pour répondre à un problème en utilisant un langage de programmation adapté ;  Inscrire l'écriture et le développement des programmes dans un travail collaboratif ;  Modéliser et simuler des phénomènes physiques, économiques et sociaux.



## Domaine 3 : Création de contenu

### 3.4 : Programmer

De quoi s'agit-il?

- 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, etc.) et pour développer un contenu riche (jeu, site web, etc.) avec des environnements de développement informatique simples, des logiciels de planification de tâches, etc...

#### Référence au socle commun :

- - Savoir que des langages informatiques sont utilisés pour programmer des outils numériques et réaliser des traitements automatiques de données (Domaine 1) ;
- - Connaître les principes de base de l'algorithmique et de la conception des programmes informatiques (Domaine 1) ;
- - Mettre en œuvre les principes de base de l'algorithmique pour créer des applications simples. (Domaine 1)

# Cadre de référence des compétences numériques : Programmer

## Niveau 1

Lire et construire un algorithme qui comprend des instructions simples.

## Niveau 2

Réaliser un programme simple.

## Niveau 3

Développer un programme pour répondre à un problème à partir d'instructions simples d'un langage de programmation ;

- Modifier un algorithme simple en faisant évoluer ses éléments de programmation ;
- Mettre au point et exécuter un programme simple commandant un système réel ou un système numérique.

## Niveau 4 (en cours)

Écrire un algorithme et développer un programme complexe pour répondre à un problème en utilisant un langage de programmation adapté ;

- Inscrire l'écriture et le développement des programmes dans un travail collaboratif ;
- Modéliser et simuler des phénomènes physiques, économiques et sociaux.

# Programmes 2016

Dans Mathématiques (Espace et géométrie)

- Au cycle 2 :

Coder et décoder pour prévoir, représenter et réaliser des déplacements dans des espaces familiers, sur un quadrillage, sur un écran. Programmer les déplacements d'un robot ou ceux d'un personnage sur un écran (Colonne 2 au cycle 2).

- Au cycle 3 :

Accomplir, décrire, coder des déplacements dans des espaces familiers.

Programmer les déplacements d'un robot ou ceux d'un personnage sur un écran (colonne 1 au cycle 3).



Quelques activités à réaliser  
au cycle 3.

Les premières activités, que l'on soit en C2 ou C3, sont celles qui mettent en oeuvre le codage de déplacement, qui permettra d'amorcer également la notion d'algorithme et de déboucher sur des constructions de figures géométriques.

### Les pré requis

Une bonne maîtrise et une bonne compréhension du repérage et du déplacement sur un quadrillage est nécessaire. (activités déjà mis en place au cycle 1)

On peut distinguer deux types d'activités

- Les activités débranchées

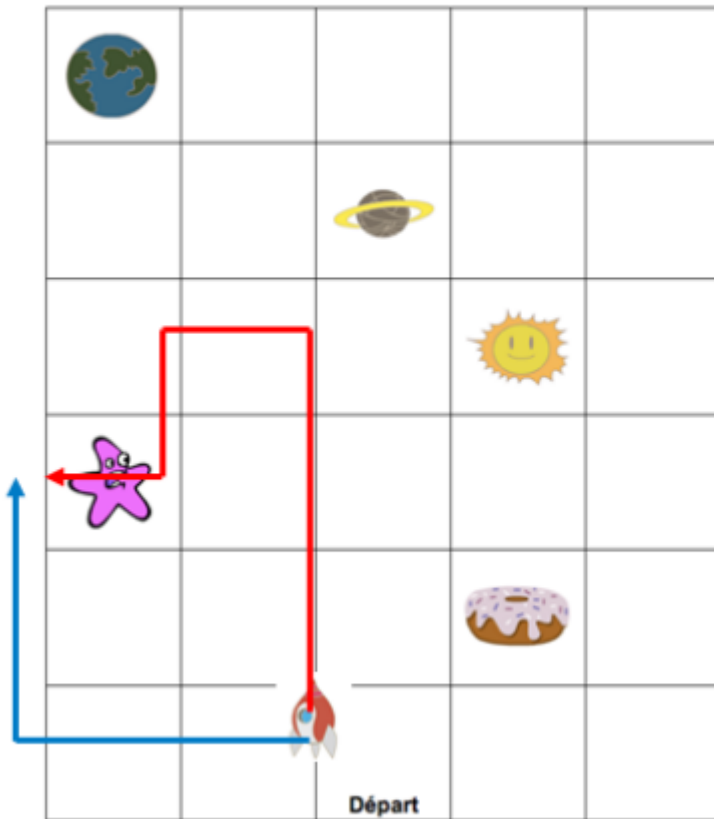
Ex : Les gobelets, la fusée, ....

- Les activités sur ordinateur, tablette,...

Activités en ligne ( Code.org par exemple) ou sur logiciel ( Scratch, Géotortue,...)

# La fusée, exemple de déplacement absolu

La fusée va se déplacer jusqu'à « l'étoile énervée ». Nous allons ici considérer deux trajets possibles.



Pour le trajet en bleu, la fusée :

- se déplace de deux cases vers la gauche ;
- monte de deux cases.

Ceci est codé à l'aide de 4 instructions écrites de gauche à droite :

**←←↑↑**

Pour le trajet en rouge, la fusée :

- monte de trois cases ;
- avance d'une case vers la gauche ;
- descend d'une case ;
- avance d'une case vers la gauche.

Ceci est codé grâce à 6 instructions :

**↑↑↑←↓←**

# Le jeu du robot

Programmer un déplacement

Lire un programme

Corriger une erreur dans un programme : déboguer





# Déplacement relatif et déplacement absolu

Il faut distinguer deux types de déplacements : le déplacement relatif et le déplacement absolu. Sans une bonne compréhension des déplacements sur un quadrillage et du repérage spatial, le déplacement relatif peut devenir une réelle difficulté pour l'élève.

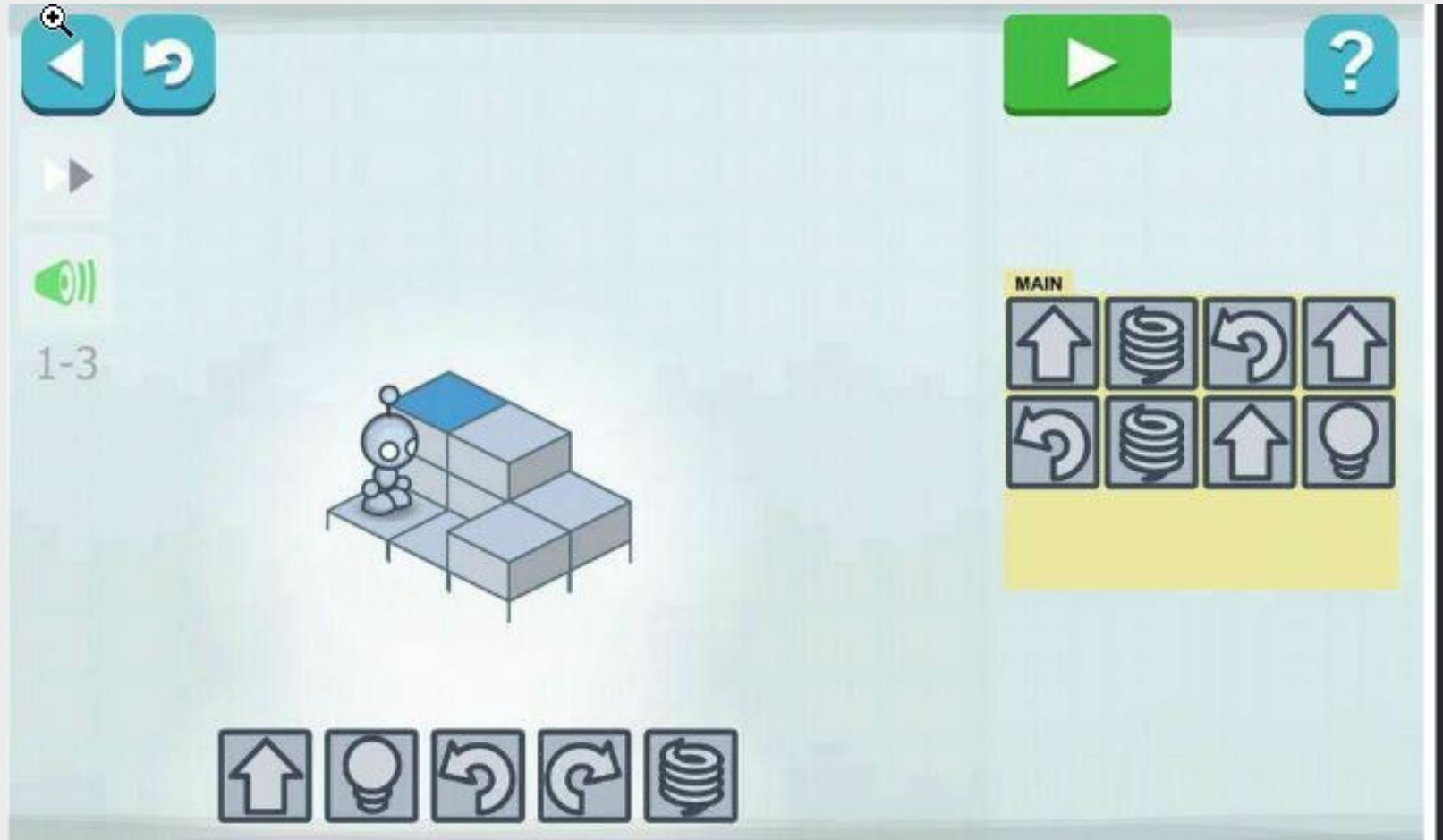
On parle de déplacement absolu, lorsque l'effet des instructions ne dépend pas de l'orientation initiale du « mobile » qui les reçoit. Par exemple : « tourne-toi vers le tableau », « avance de deux pas vers la fenêtre », « avance de trois pas vers l'est » Ex : La fusée

On parle de déplacement relatif lorsque l'effet des instructions dépend de l'orientation initiale du « mobile » qui les reçoit. Par exemple : « effectue un quart de tour vers la droite », « avance de trois pas ». Ex : Lightbot ou le jeu de l'abeille.

*Initiation à la programmation aux cycles 2 et 3*

[eduscol.education.fr/ressources-2016](https://eduscol.education.fr/ressources-2016)

# Lightbot, exemple de déplacement relatif



Lightbot est disponible sous Android et IOS.

# Jeu de l'abeille sur Code.org, un autre exemple de déplacement relatif



Étape 8: Abeille: Boucles

7



Démarrer

Étape



Chaque fleur contient une unité de nectar. Récolte le

Blocs

Espace de

avancer plus

quand l'exécution commence

tourner à gauche ↶

tourner à droite ↷

obtenir du nectar

fabriquer du miel

répéter ??? fois

faire

# Les gobelets : une activité débranchée