



A collage of three images: a close-up of a metal pipe and valve on the left, a view of industrial shelving units in the middle, and a robotic arm with a camera and sensor on the right.

# DOCUMENT D'ACCOMPAGNEMENT ACTIVITÉ PROGRAMMER AU VAISSEAU

Arnaud Grandadam, Julien Narboux et Basile Sauvage

UFR

de mathématique  
et d'informatique

Université de Strasbourg



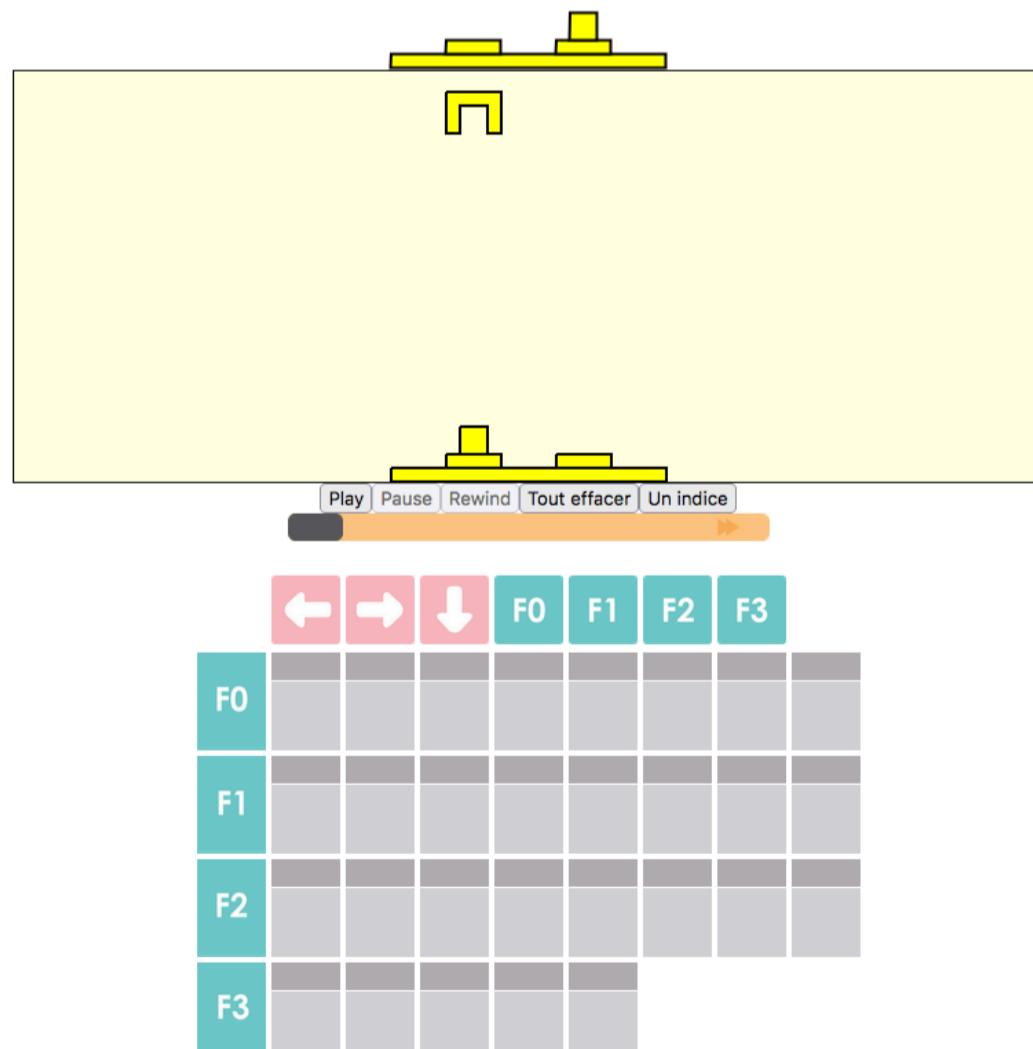
<b>Introduction.....</b>	<b>3</b>
<b>Contexte pédagogique.....</b>	<b>4</b>
Adéquation aux programmes .....	4
Informatique débranchée .....	5
Informatique tangible .....	5
<b>Contexte scientifique .....</b>	<b>6</b>
<b>Activités connexes .....</b>	<b>7</b>
<b>Contacts.....</b>	<b>8</b>

# INTRODUCTION

Ce document, rédigé à l'attention des enseignants, a pour but de donner les éléments de contexte pédagogique et scientifique permettant d'appréhender le dispositif muséographique **Programmer** de l'exposition permanente La Fabrique, ainsi que de fournir des liens vers d'autres activités pouvant être réalisées en classe. Le public visé est les élèves de fin de cycle 1, cycle 2 ou cycle 3.

L'activité est proposée en accès libre dans l'exposition. Il existe une version étendue, qui ajoute de nouvelles instructions et des défis plus compliqués. Cette version est accessible sous forme d'atelier animé par un médiateur scientifique au Vaisseau, et librement sous forme numérique en ligne.

Le dispositif muséographique présenté au Vaisseau est inspiré du jeu [Cargo-Bot](#) développé par Rui Viana sur IPad, puis adapté en des versions libres disponibles en ligne ou sans ordinateur par le groupe [InfoSansOrdi](#).



<https://www-verimag.imag.fr/~wack/CargoBot/>

# **CONTEXTE PÉDAGOGIQUE**

## **ADÉQUATION AUX PROGRAMMES**

Apprendre les bases de l'algorithmique et de la programmation entre complètement dans les programmes de l'école élémentaire. Il s'agit surtout, à ce niveau, d'une initiation qui permet de développer des savoirs et des savoirs-faire. Ainsi, savoir décomposer un problème en tâches simples, reconnaître des tâches que l'on a déjà effectué ou qui se répètent, apprendre à travailler ensemble, favoriser l'imagination et le sens créatif sont quelques-unes des compétences à l'œuvre. Elles sont également mobilisées dans d'autres enseignements : en mathématiques bien entendu, mais également en sciences, ou même en production de textes.

Les compétences à mettre en œuvre au cycle 2 et au cycle 3 sont détaillées notamment dans le thème espace et géométrie : se repérer et se déplacer en utilisant des repères et des représentations ; coder et décoder pour prévoir de représenter et réaliser des déplacements dans des espaces familiers, sur un quadrillage ou sur un écran. Dans l'activité en accès libre au Vaisseau, l'élève investira les relations entre l'espace dans lequel on se déplace et ses représentations, il devra anticiper et coder une suite de déplacements. Dans l'activité étendue (version avec atelier animé au Vaisseau, et [version numérique en ligne](#)), l'élève explore également la décomposition en sous-problèmes et la répétition de tâches, par le biais d'une nouvelle instruction « fonction ».

Le cycle 1 n'est pas en reste. Ce qui est attendu des enfants en fin d'école maternelle se décline ainsi :

- Situer des objets par rapport à soi, entre eux, par rapport à des objets repères.
- Se situer par rapport à d'autres, par rapport à des objets repères.
- Dans un environnement bien connu, réaliser un trajet, un parcours à partir de sa représentation (dessin ou codage).
- Utiliser des marqueurs spatiaux adaptés (devant, derrière, droite, gauche, dessus, dessous, etc.) dans des récits, descriptions ou explications.

Ressources :

- [Programmes du cycle 1](#) D'après le BOEN n° 31 du 30 juillet 2020  
(échanger et réfléchir avec les autres page 9, codage pages 29-30).
- [Programmes du cycle 2](#) D'après le BOEN n° 31 du 30 juillet 2020  
(argumenter pages 5-6, déplacement dans l'espace et robots pages 50, 62).
- [Programmes du cycle 3](#) D'après le BOEN n°31 du 30 juillet 2020  
(argumenter 18, 47, 48, programmation, robotique page 97).

## **INFORMATIQUE DÉBRANCHÉE**

L'informatique débranchée est une méthode d'enseignement des concepts et principes informatiques sans l'utilisation d'ordinateurs ou d'autres appareils électroniques. Elle repose sur l'idée que les élèves peuvent apprendre d'importants concepts et principes informatiques par le biais d'activités et de jeux conçus pour être attrayants et amusants. Ces activités impliquent souvent l'utilisation d'objets physiques, tels que des cartes ou d'autres outils de manipulation, pour représenter des données et des algorithmes.

En utilisant des activités et des jeux pratiques, les élèves peuvent apprendre des concepts informatiques importants sans se sentir dépassés ou intimidés et sans être limités par les aspects techniques. Un autre avantage de l'informatique débranchée est qu'elle offre aux élèves la possibilité de développer d'importantes compétences en matière de résolution de problèmes et de pensée critique. En travaillant sur les activités et les jeux, les élèves sont mis au défi de penser de manière créative et logique afin de résoudre des problèmes et de comprendre des concepts complexes. Cela peut aider à développer des compétences qui sont applicables non seulement en informatique, mais aussi dans de nombreux autres domaines d'étude et dans des situations du monde réel.

Ressources:

- L'informatique débranchée, Magazine Tangente, [numéro 147](#).
- [Un livre sur l'informatique débranchée](#)

## **INFORMATIQUE TANGIBLE**

La programmation tangible est une méthode d'enseignement et d'apprentissage de la programmation informatique utilisant des objets ou des dispositifs physiques comme moyen de représenter et de manipuler le code. Cela peut impliquer l'utilisation d'outils tangibles tels que des blocs ou des cartes sur lesquels sont imprimées différentes instructions, qui peuvent être arrangées et combinées pour créer un programme. L'idée de la programmation tangible est de rendre le processus d'apprentissage du code plus concret, ce qui facilite la compréhension et l'engagement des élèves. Des exemples d'activités tangibles sont proposés ci-dessous.

# CONTEXTE SCIENTIFIQUE

Cette activité permet d'aborder plusieurs concepts de la science informatique. Il s'agit d'écrire un programme qui sera lu et exécuté par un ordinateur pour commander un robot. La robotique rend l'activité ludique et attrayante, mais c'est bien de l'informatique qui est vulgarisée ici, à travers une activité de découverte de la programmation.

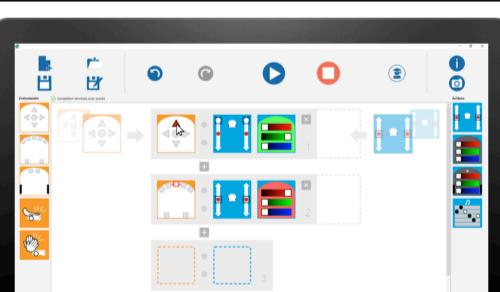
Voici quelques définitions du vocabulaire, et une manière de le relier à l'activité :

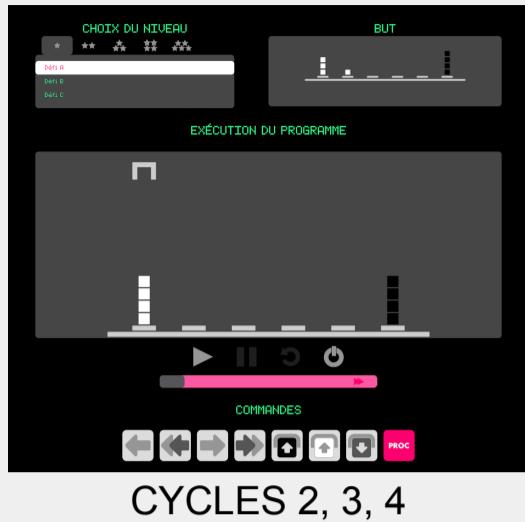
- **Algorithme** : suite organisée d'instructions, permettant de résoudre un problème. Un algorithme peut être décrit de différentes manières, par exemple à l'écrit, à l'oral, ou à l'aide d'un schéma. Ainsi, la technique opératoire pour faire une addition à plusieurs chiffres est un algorithme. Dans l'activité **Programmer**, le problème à résoudre est de déplacer les cubes pour passer d'une configuration initiale (plateau vide) à une configuration finale.
- **Programme** : écriture d'un algorithme à l'aide d'un langage informatique, permettant son exécution par un ordinateur. Le mot « code » est parfois utilisé comme synonyme de « programme ». La distinction entre algorithme (l'idée) et programme (sa réalisation) a peu d'importance dans le cas présent : nous recommandons ici d'utiliser le mot « programme ».
- **Instruction** : commande simple qui peut être exécutée. Dans le cas présent, les instructions sont représentées par les tuiles (déposer un cube, se déplacer, appeler la fonction, etc.).
- **Fonction** : groupe d'instructions qui effectue une tâche spécifique dans un programme. Il y a la « définition » de la fonction, qui décrit une fois pour toutes le groupe d'instructions. Il peut y avoir plusieurs « appels » de la fonction, chaque appel commandant d'exécuter tout le groupe d'instructions. Dans d'autres activités, on parle parfois de « procédure » (avec Lightbot par exemple).
- **Bug** : erreur de conception dans un programme. Dans cette activité on découvre deux types de bugs : les instructions impossibles à exécuter (par exemple poser un cube alors que la pince du robot est vide), et les suites d'instructions qui ne permettent pas d'atteindre l'objectif (par exemple, poser un cube au mauvais endroit). La détection de bugs est une tâche très difficile, même pour des informaticiennes et informaticiens experts.

Ressources :

- Un épisode de la série des Sépas illustre la notion de bug en la replaçant dans l'évolution de l'informatique <https://www.youtube.com/watch?v=8-oUkA9ScsY>
- L'origine du mot « algorithme » <https://www.youtube.com/watch?v=QbZms6RgLRE>

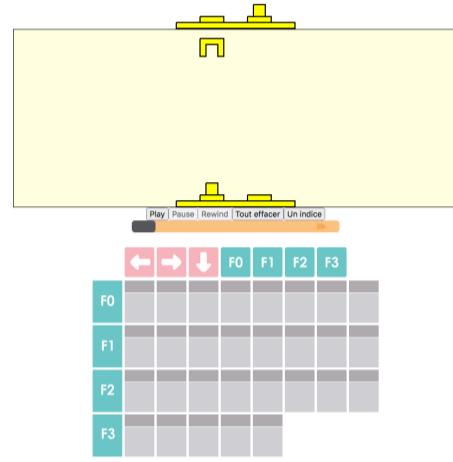
# ACTIVITÉS CONNEXES

 <p>CYCLE 1</p>	<p>Mettre en œuvre des activités de codage séquentielles pour le cycle 1 en branché ou débranché (avec une partie tangible) pour permettre aux enfants d'apprendre à intégrer progressivement la chronologie des tâches requises et à ordonner une suite d'actions.</p> <ul style="list-style-type: none"> <li>- construire le nombre</li> <li>- s'orienter / se latéraliser</li> <li>- travailler le langage</li> </ul>	<a href="#">Lien 1</a> <a href="#">Lien 2</a>
 <p>CYCLES 2 &amp; 3</p>	<p>Activités branchées ou débranchées, tangibles : construire des défis pour amener les élèves à anticiper une position, argumenter, optimiser leur séquence.</p>	<a href="#">Lien</a>
 <p>CYCLES 3 et 4</p>	<p>Entrer dans la programmation événementielle avec Thymio.      Introduire le SI ... Alors      "Coder et décoder pour prévoir"</p>	<a href="#">Lien</a>



Une version en ligne du jeu avec les mêmes instructions et défis que ceux proposés au Vaisseau.

[Lien](#)



Les versions branchée et débranchée de l'activité proposée par l'**IREM de Grenoble**

[Lien](#)

[Lien](#)

CYCLES 3, 4



Une autre version débranchée de l'activité proposée par l'**IREM de Clermont Ferrand**

[Lien](#)

CYCLES 3 et 4

## CONTACTS

Les ERUN (Enseignants Référents aux Usages du Numériques) ou les CPD (Conseillers Pédagogiques Départementaux) sont à votre disposition pour toute question ou demande d'aide à la mise en œuvre des projets autour de l'algorithme et du codage.  
Contact : → [arnaud.grandadam@ac-strasbourg.fr](mailto:arnaud.grandadam@ac-strasbourg.fr)