

# DÉFIS BLUEBOT

Document d'accompagnement



## Présentation

Ces défis de **difficulté progressive** vont permettre aux élèves de s'initier au codage et à la programmation à l'aide de robots BlueBots (ou BeeBots).

Ce robot utilise des **déplacements relatifs** avec des rotations de 90° (par opposition aux déplacements absolus de type Nord, Sud, Est, Ouest).

Ces défis permettront aux élèves de :

- **programmer** le déplacement d'un robot sur un quadrillage,
- **anticiper** le déplacement du robot avant de le vérifier avec l'appareil,
- **décomposer** un problème complexe en problèmes simples,
- **corriger** un bug dans un programme,
- **comparer** plusieurs solutions à un même problème et en dégager la solution optimale (programme utilisant le moins d'instructions possibles),
- **coopérer** pour résoudre une tâche complexe.

## Organisation pédagogique

**Avant de commencer** cette activité, il est nécessaire de faire vivre avec les élèves **une situation de numérique débranché** utilisant des déplacements relatifs, par exemple le « Robot idiot ».

### MATERIEL A PREVOIR

- Une **grille 4 x 5** pour le déplacement des robots (le quadrillage sera formé de lignes espacées de **15 cm** ce qui correspond exactement au « pas » du robot).
- Les **fiches défis en A4** plastifiées pour être utilisées plusieurs fois.
- Un **carnet individuel agrafé** regroupant les fiches défis au format A5 pour le suivi des élèves.

### MISE EN ŒUVRE

En préalable aux défis, il est indispensable de **présenter le robot** et son fonctionnement.

L'ensemble des défis peut être réalisé **individuellement ou en binôme** (en atelier ou avec la grille et le robot à disposition dans un coin de la classe).

Pour la plupart des défis, une **mise en commun** permet de revoir avec les élèves les points importants, tels qu'indiqués dans la grille de description des défis.

Dans un **premier temps**, les élèves préparent les programmes à l'aide des fiches défis.

### Recommandations

- Imposer d'écrire le programme sur le livret avant de manipuler le robot.
- Ne pas autoriser le déplacement autour du quadrillage : les élèves sont placés face au quadrillage, dans le sens du livret.
- Ne pas déplacer le robot manuellement pendant l'écriture du programme.




Dans un **second temps**, les élèves testent leurs programmes avec le robot sur la grille grand format.

### **Pistes de différenciation**

- Tant que la touche « Efface » n'est pas pressée (ou que le robot n'est pas éteint), le robot conserve en mémoire les instructions : il est donc possible de programmer par étapes.
- Autoriser le déplacement des élèves autour du quadrillage pour faciliter la programmation en déplacements relatifs en s'orientant comme le robot.
- Autoriser le déplacement manuel du robot pour permettre la décomposition étape par étape lors de l'écriture du programme et ne pas imposer l'anticipation.
- Du matériel papier peut également être proposé : une grille de test, des abeilles en miniatures, les instructions (ce matériel est à disposition dans ce document).

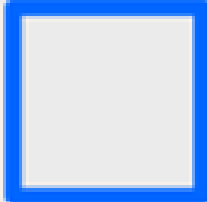

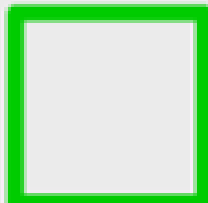


### **PROLONGEMENTS POSSIBLES**

- Condamner une case.
- Ajouter des barrières entre certaines cases.
- Faire écrire des défis aux élèves.
- Créer de nouvelles pistes...

Défi	Intentions pédagogiques
1	Prise en main du robot. Utilisation de l'instruction « avance » et de la touche « Go ».
2	Suite de la prise en main. Utilisation d'une nouvelle instruction « tourner à... » et de la touche « Go ». Introduction du déplacement relatif.
3	Respect d'une contrainte (réinvestissement).
4	Optimisation d'un programme. Introduction de l'instruction « recule ». Comparaison de programmes fonctionnels pour mettre en valeur celui qui utilise le moins d'instructions : 
5	Difficulté à coder le déplacement du robot lorsqu'il est orienté vers l'élève.
6	Réinvestissement de l'ensemble des acquis pour réussir une mission plus complexe. Comparaison des multiples programmes. Recherche du programme optimal.
7	Introduction de la notion de bug. Repérage et correction d'une erreur (éventuellement avec aide du robot). Plusieurs possibilités de débogage : 
8	Réinvestissement des acquis avec une contrainte forte : une instruction interdite.
9	Réinvestissement des acquis avec une contrainte forte : une instruction interdite.
10	Repérage et correction d'un bug (éventuellement avec aide du robot). Difficulté à coder le déplacement du robot lorsqu'il est orienté vers l'élève. Plusieurs possibilités de débogage : 
11	Tâche complexe avec position de départ imposée. Comparaison de programmes, recherche de l'optimisation.
12	Réinvestissement des acquis avec une contrainte forte : deux instructions interdites.
13	Tâche complexe avec choix libre du départ. Comparaison de programmes, recherche de l'optimisation.
14	Coordination de deux robots.
15	Coordination de deux robots. Possibilité d'utiliser l'instruction « pause » (attention, une seule pause ne suffit pas).
16	Coordination de deux robots. Recherche de l'optimisation.

# Défis BlueBot : grille de test

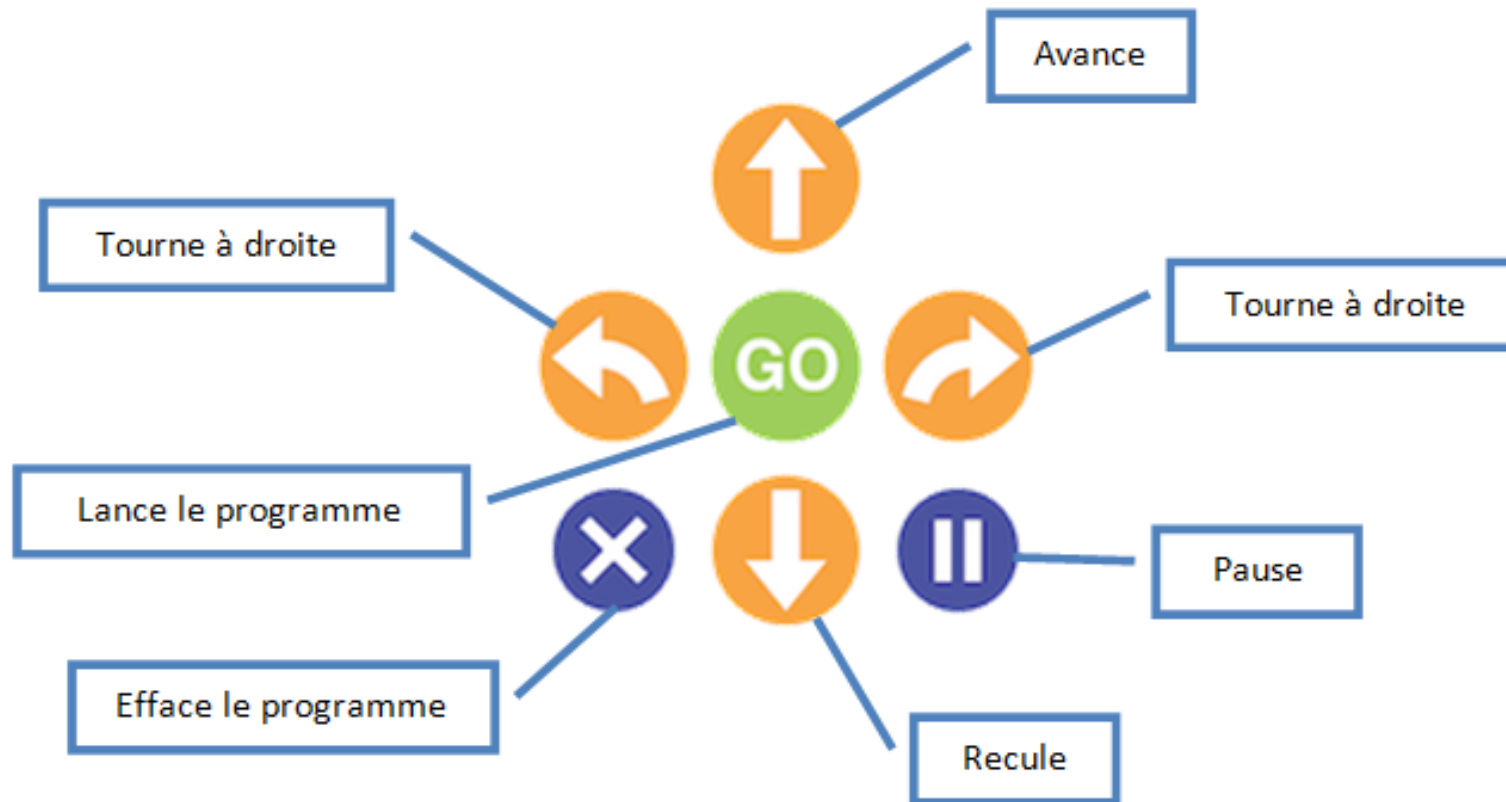


Ces robots peuvent être découpés et plastifiés pour être utilisés avec la fiche de test.



# Instructions du robot BlueBot



Avant chaque nouveau défi, pense à effacer le programme précédent à l'aide de la touche :

