

Séquence 4

Comment optimiser le déplacement d'un robot ?

CT 2.7

Imaginer, concevoir et programmer des applications informatiques nomades.

Activité 1 : Comment faire suivre une ligne à un robot ?

Objectifs de l'activité :

- Comprendre et utiliser des variables statiques et dynamiques
- Programmer le robot pour qu'il réagisse à des situations précises

1) Situation déclenchante (10 Minutes)

En utilisant le texte de la situation déclenchante présente sur le site Mr Coppin ci-dessus, **réponds** aux questions :

✓ D'après-toi, que signifie « trajet aléatoire » ?

✓ Le robot peut-il sortir de la zone de tonte ? Pourquoi ? Qu'est-ce-qui l'en empêche ?

✓ D'après-toi, comment l'Automower fait-il pour suivre le câble guide ? Par quel type de capteur ?

2) Simulation avec le mBot - Analyse du comportement : (15 minutes)

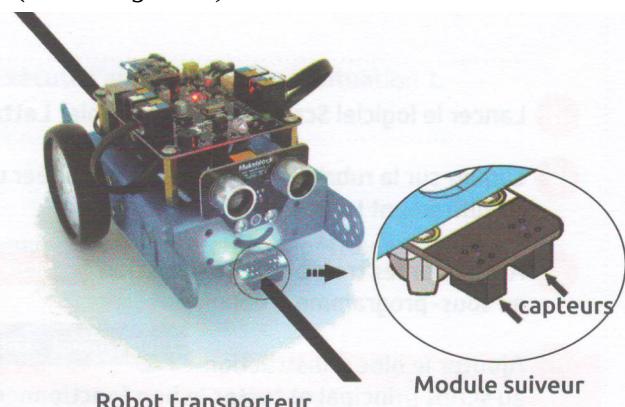
Mark aimerait dans un premier temps que son robot le suive dans sa tâche de découverte de la planète pour transporter le matériel scientifique et la nourriture. Pour cela il a eu l'idée de placer des lignes sur le sol derrière lui pour permettre au robot de suivre sa trace à l'aide de son capteur suiveur de ligne.

Consigne : Indiquez dans le tableau la position des capteurs (droite et gauche) dans les situations 2, 3 et 4 :

Le robot transporteur emmène des objets d'un point à un autre en suivant un marquage au sol (ligne noire). Pour assurer cette fonction, il dispose à l'avant d'un **module suiveur de ligne**, composé de deux capteurs optiques.

Tant que les deux capteurs détectent la ligne, le robot avance (**situation 1**). Lorsqu'un des deux capteurs ne détecte plus la ligne, le robot tourne sur lui-même pour se remettre dans l'axe (**situation 2 ou 3**). Si les deux capteurs sont en dehors de la ligne, le robot recule (**situation 4**).

● **Comment programmer le robot pour qu'il suive la ligne noire ?**



Robot transporteur

Module suiveur

Comportement du robot

	Situation 1 Le robot avance	Situation 2 Le robot tourne à gauche	Situation 3 Le robot tourne à droite	Situation 4 Le robot recule
➤ Le capteur droit est sur la ligne.	➤	➤	➤	➤
➤ Le capteur gauche est sur la ligne.	➤	➤	➤	➤
Valeur capteur suiveur	0	1	2	4

3) Simulation avec le mBot - Modifier, compléter, créer l'algorithme : (10 minutes)

- ✓ Dans le tableau ci-dessous, complète chaque test de l'algorithme pour les situations 2 et 3 avec les valeurs du capteur suiveur de la dernière ligne du tableau précédent. Ces tests permettent au robot de se remettre dans l'axe de la ligne noire :

Situation 1 Les deux capteurs détectent la ligne noire	Situation 2 Le capteur droit ne détecte plus la ligne noire	Situation 3 Le capteur gauche ne détecte plus la ligne noire
Si État suiveur de ligne = 0 Alors Faire avancer le robot Sinon	Si État suiveur de ligne = 1 Alors ... Sinon	Si État suiveur de ligne = ... Alors ... Sinon

- ✓ Que fait le robot si l'état du module suiveur de ligne est différent de 0, 1 ou 2 ? _____

4) Simulation avec le mBot - Écrire un programme et l'exécuter : (durée : 40 minutes)

A partir de l'écran ci-dessous qui décrit le début du programme, réponds aux questions suivantes. Lis bien et prends le temps de comprendre l'encadré « information ci-dessous »

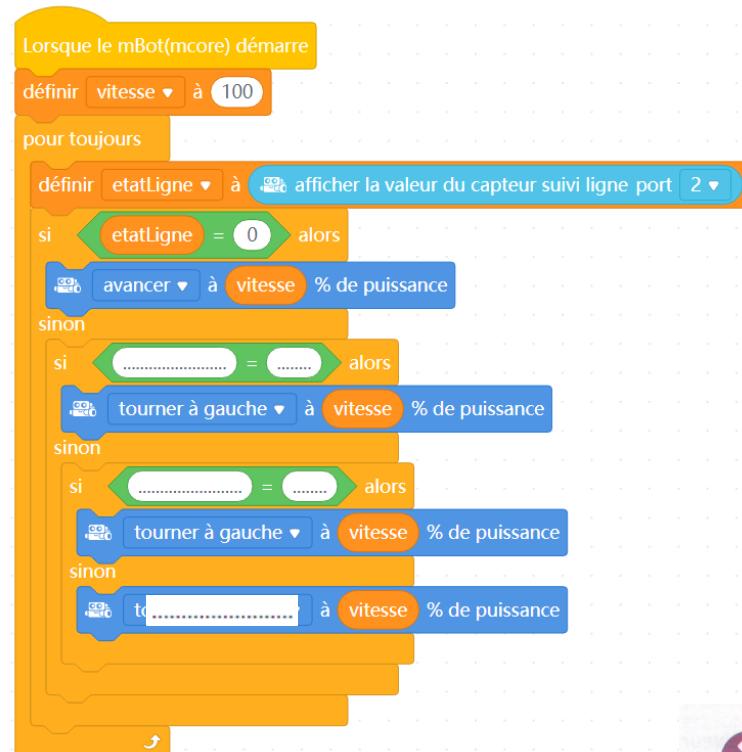


- ✓ Quel est le rôle du bloc d'instructions suivant ?



- ✓ Quel est le numéro de port utilisé pour récupérer l'état du module suiveur de ligne ? _____

- ✓ Explique le fonctionnement du bloc d'instructions « Si etatLigne = 0 Alors avancer à vitesse % de puissance » :



- ✓ Écris la fin du programme ci-dessous pour que le robot puisse tourner (situation 2 et 3) et reculer (situation 4).
✓ Lance le logiciel mBlock et ouvre le fichier **Robot suiveur** (sur le site de Mr Coppin)
✓ Modifie le fichier à l'aide des blocs de programmation comme tu l'as indiqué ci-dessus
✓ Allume le robot, connecte-le à l'ordinateur et implante le programme en suivant les instructions dans la fiche-méthode : **Fiche Méthode mBot**.
✓ Teste le robot sur la piste présente dans la boîte du robot et vérifie le bon fonctionnement du module suiveur de ligne. Corrige le programme et teste-le encore si nécessaire.

Aller plus loin

Proposer une amélioration du programme pour que le robot s'arrête 20 cm avant de percuter un obstacle.

