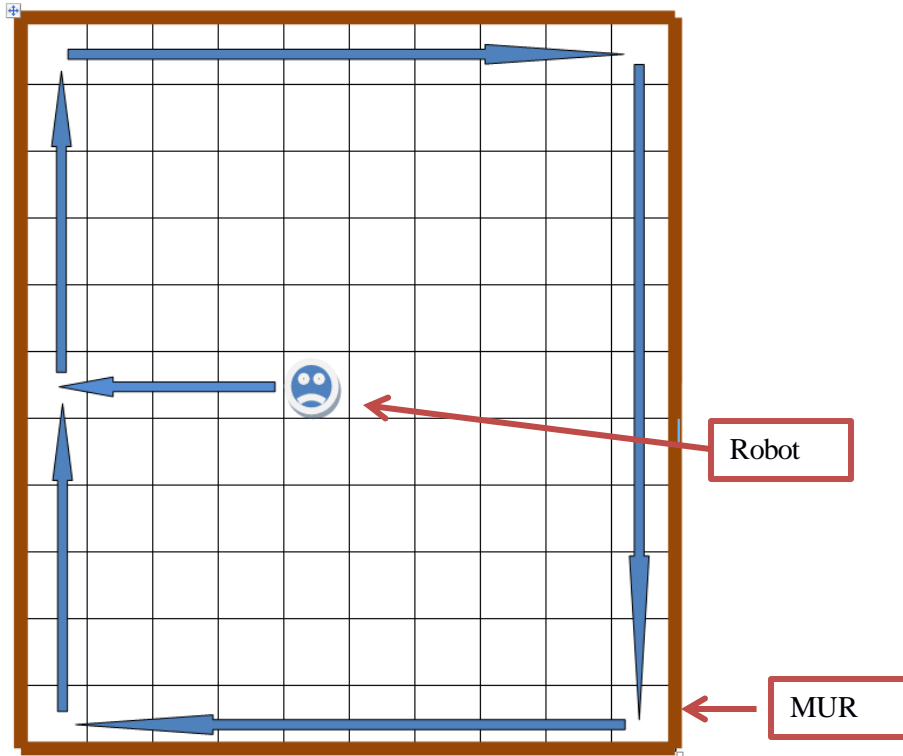


Programmation

Exercice 1 : Ecrire le programme permettant au robot de suivre les trajectoires des flèches. Sous forme de logigramme.



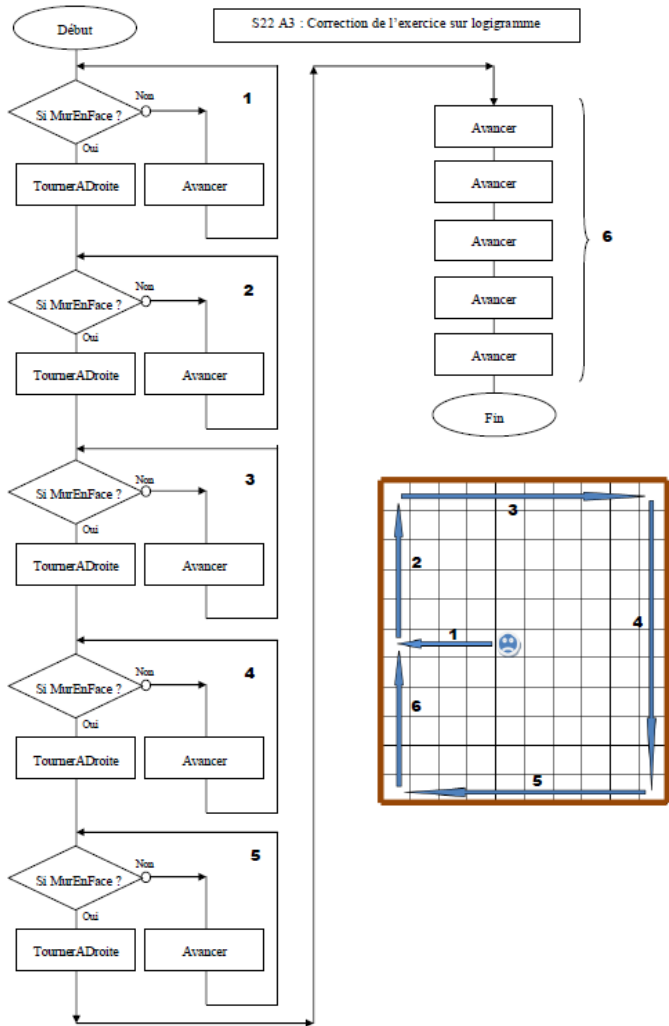
Liste des instructions :

Actions :

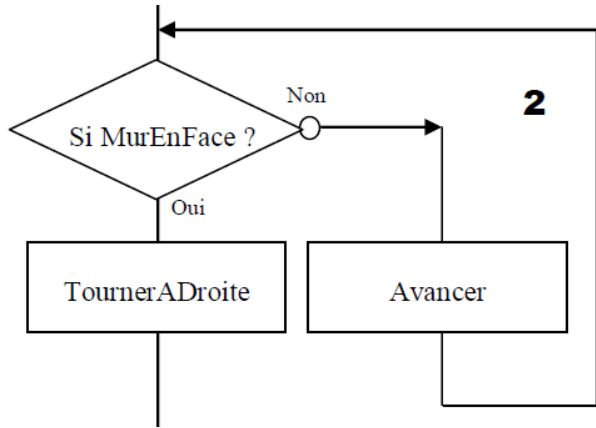
- Avancer : fait avancer le robot d'une case
- TournerADroite : Oriente le robot à droite de 90°
- TournerAGauche : Oriente le robot à gauche de 90°

• Conditions :

- MurEnFace : Permet au robot de détecter un mur en face de lui.



Correction

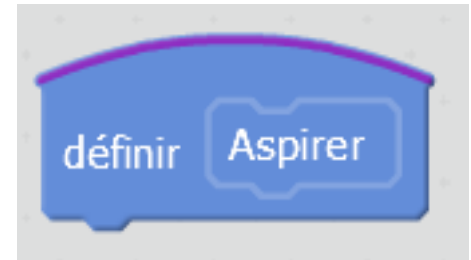
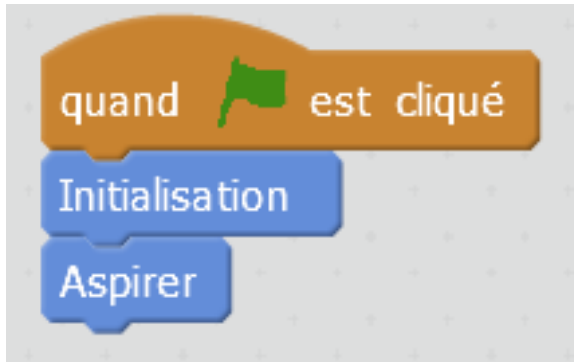


Exercice 2 : Programmez un déplacement aléatoire avec scratch.

1. Lancez scratch et **choisissez** comme lutin *Button1*
2. Mettez la taille de **Button1** à 80%
3. Placer Button1 en bas à gauche (Origine de départ de votre robot)
4. Ecrire un programme permettant à **Button1** de se déplacer de manière aléatoire à la **vitesse de 10** pour simuler le nettoyage d'une pièce.
5. Lors du déplacement du lutin laisser une trace écrite au sol de **largeur 20**
6. **Placez** un chronomètre pour mesurer en combien de temps votre pièce est nettoyée ?

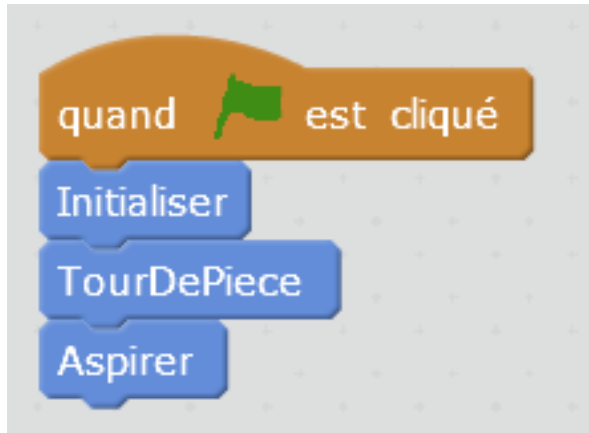
Exercice 3 : Programmez des fonctions.

1. **Modifiez** votre programme en ajoutant les fonctions initialiser et aspirer dans votre programme principal.

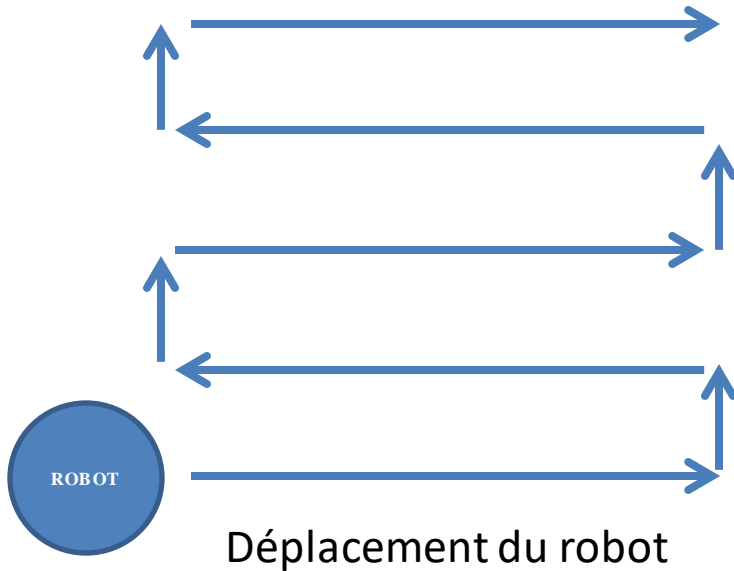


Exercice 4 : Programmez un tour de pièce + un déplacement aléatoire avec scratch.

1. **Modifiez** votre programme en ajoutant la fonction *tourDePièce* pour que votre programme face un tour complet de la pièce avant de se déplacer de manière aléatoire.

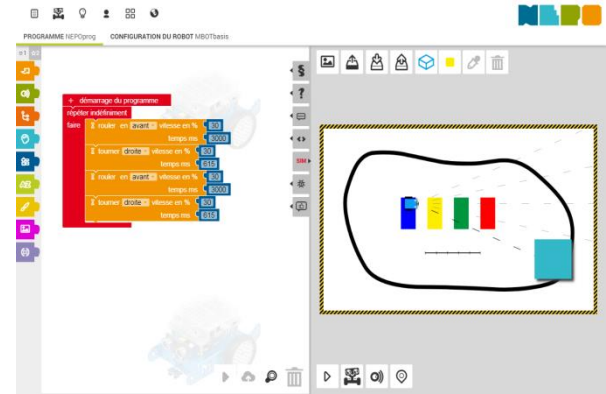


Exercice 5 : Programmez des déplacements .



1. **Créez** un nouveau programme qui parcourt la pièce en faisant des allers / retours. Quand le robot arrive en haut il retourne dans le coin inférieur gauche.

```
+ démarrage du programme
répéter indéfiniment
faire
  rouler en avant vitesse en % 30
  temps ms 3000
  tourner droite vitesse en % 30
  temps ms 615
  rouler en avant vitesse en % 30
  temps ms 3000
  tourner droite vitesse en % 30
  temps ms 615
```



EXERCICES



4èmes

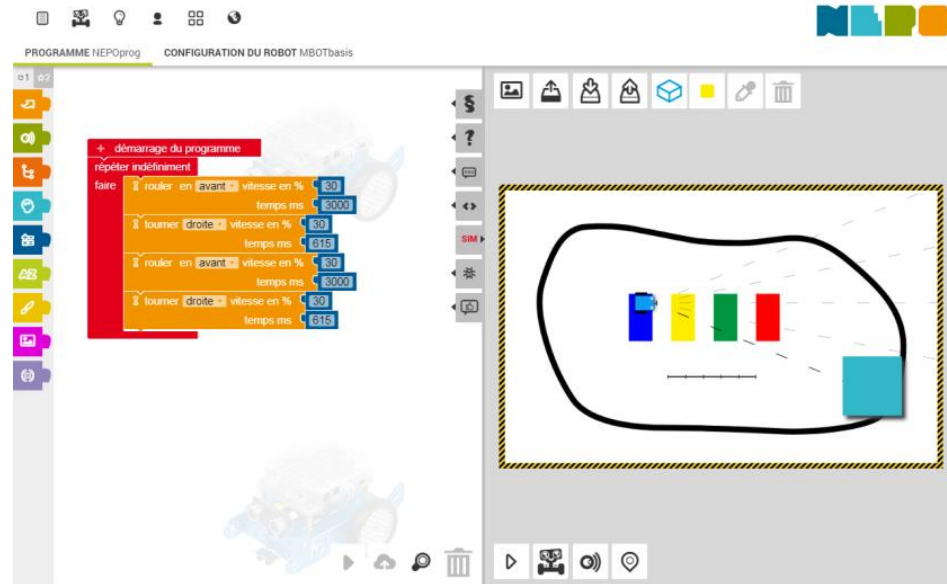
3èmes



Simulation du robot mBot



- Pour programmer notre robot mBot nous allons utiliser un logiciel de simulation



<https://lab.open-roberta.org/>



Programme

Instructions

+ démarrage du programme

répéter indéfiniment

faire

- rouler en avant vitesse en % 30 temps ms 3000
- tourner droite vitesse en % 30 temps ms 615
- rouler en avant vitesse en % 30 temps ms 3000
- tourner droite vitesse en % 30 temps ms 615

The simulation area features a central track with a blue robot, a yellow obstacle, a green obstacle, and a red obstacle. A cyan square is positioned at the end of the track. The track is enclosed in a yellow and black striped border. Above the track is a toolbar with icons for image, upload, save, home, cube, yellow square, eraser, and delete. To the left of the track is a vertical toolbar with icons for dollar sign, question mark, speech bubble, double arrows, SIM button, and other controls.

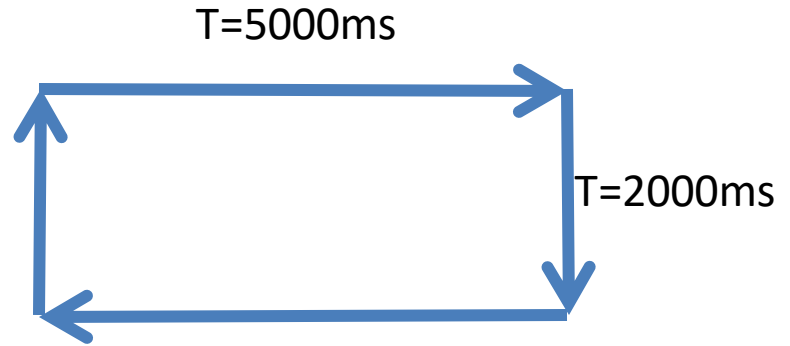
Mode simulation

Réinitialiser le robot

Exécuter la simulation

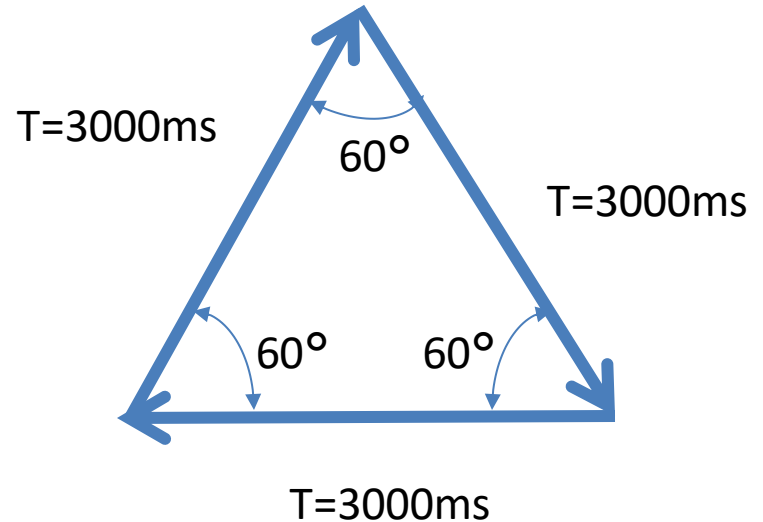
Exercice 1 (Difficulté *)

- Vous devez dessiner un rectangle avec votre robot



Exercice 2 (Difficulté **)

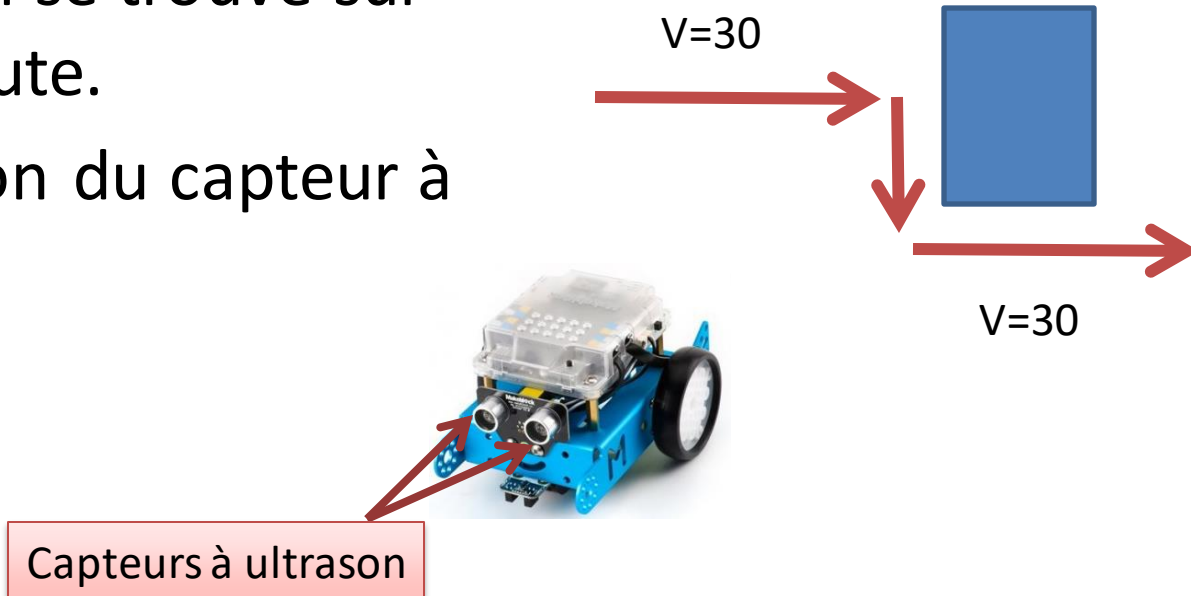
- Vous devez dessiner avec votre robot mBot un triangle équilatéral (3 cotés égaux)



Exercice 3 : Eviter un obstacle

(Difficulté **)

- Vous devez éviter un objet qui se trouve sur votre route.
- Utilisation du capteur à ultrason



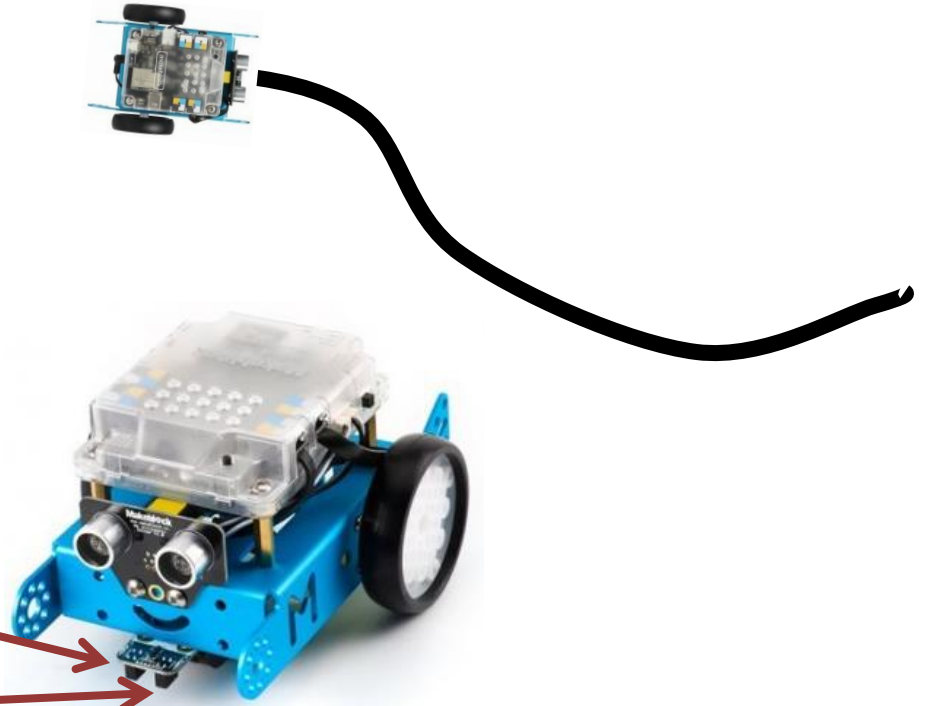
Exercice 4 : suivre une ligne

(Difficulté ***)

- Vous devez éviter un objet qui se trouve sur votre route.
- Utilisation du capteur de détection de ligne noire

Capteur ligne noire droite

Capteur ligne noire gauche



Exercice 4 : suivre une ligne

(Difficulté ***)

| Capteur | Noir | Blanc |
|---------|----------------|-----------------|
| Right | True (Vrai) | False (Faux) |
| Left | True (Vrai) | False (Faux) |

- Si le capteur (Droite) (Right) détecte du **noir**
 - il retournera « false" ou « faux"
 - il retournera "true" ou "vrai" s'il détecte du blanc
- Même fonctionnement pour le capteur (Gauche) (Left)

FIN