

Je détecte des obstacles et je mesure une distance

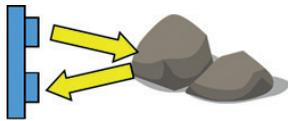
Ton robot Geek Junior est équipé d'un capteur à ultrasons. Celui-ci va te permettre de mesurer des distances.

Attention : tout capteur et actionneur possède une plage d'utilisation, c'est-à-dire des valeurs selon lesquelles le capteur/actionneur peut et doit fonctionner. En dehors de ces valeurs, il se produit un dysfonctionnement ou un non-fonctionnement du système.

La plage d'utilisation pour notre capteur à ultrasons se situe entre 2 cm et 350 cm. Ceci veut dire que le capteur n'est pas capable de mesurer une distance inférieure à 2 cm et supérieure à 350 cm. Il convient de le prendre en compte dans tes manipulations.

1

Un peu de théorie



Le capteur envoie une onde à l'aide de sa broche Tx, cette onde ricoche contre un obstacle et se retrouve captée par Rx du capteur. C'est donc la vitesse que met l'onde pour effectuer l'aller-retour qui permet de déterminer la distance. Malin!

2

Mesurer la distance

Nous allons faire mesurer la distance entre le capteur ultrason de ton robot et un obstacle ET faire afficher celle-ci sur la matrice Del.

À l'aide des instructions ci-contre, réaliser le script qui répondra au fonctionnement suivant : lorsque le bouton **A** de la carte **micro:bit** a été actionné, il faut faire afficher la distance en cm sur la matrice Del.

```

lorsque le bouton A est pressé
  montrer nombre 0
  Ultrasonic Sensor (in cm) at P8
  
```

Mais comment trouver le bloc « **Ultrasonic Sensor** » ? Direction le menu « **Grove** ».

```

lorsque le bouton A est pressé
  montrer nombre Ultrasonic Sensor (in cm) at P8
  
```

3

Afficher la valeur sur la carte micro:bit

Sur **MakeCode**, commence par écrire ce petit script :

```

toujours
  montrer nombre Ultrasonic Sensor (in cm) at P8
  
```

Charge ensuite le fichier correspondant à ce script sur ta carte **micro:bit**. Tu as oublié comment faire ? Retourne au tuto 1 de ce cahier d'activités.



De quelle différence t'aperçois-tu ? Normalement, tu dois observer que la valeur mesurée change sans arrêt et que le résultat est très difficile à lire.

Comment faire pour remédier à ce problème ? Très simple : il faut modifier le script comme ci-dessous et charger une nouvelle fois le fichier modifié sur la carte **micro:bit**.

```

toujours
  montrer nombre Ultrasonic Sensor (in cm) at P8
  pause (ms) 2000
  
```

Tu peux utiliser aussi le fichier exemple « **robot_geek_tuto5_2.hex** » que tu peux télécharger depuis cette adresse: <https://bit.ly/robotgeekjunior> pour retrouver ce script.

4

Écrire le script pour que le robot s'arrête

Tu sais donc maintenant comment procéder pour détecter un obstacle et mesurer une distance entre cet obstacle et ton robot.

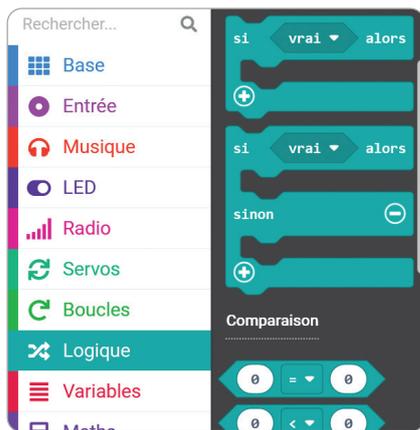
À partir de là, on va utiliser le script **robot_geek_tuto2_1.hex** (cela te rappelle le tuto n° 2, non ?).

```

au démarrage
  servomoteur P0 à rotation continue fonctionne à 0 %
  servomoteur P1 à rotation continue fonctionne à 0 %

toujours
  servomoteur P0 à rotation continue fonctionne à 50 %
  servomoteur P1 à rotation continue fonctionne à -50 %
  pause (ms) 1000
  
```

Essaie de modifier le script avec les blocs de commande ci-dessous :



L'objectif est d'indiquer au robot: « Si la distance mesurée est inférieure à 20 cm ALORS le robot s'arrête SINON il avance. »

Tu as réussi à bien imbriquer les blocs de commandes? Tu n'y arrives pas? Pas de panique, télécharge le fichier **robot_geek_tuto5_3.hex** que tu peux télécharger depuis cette adresse: <https://bit.ly/robotgeekjunior>.

```

au démarrage
  servomoteur P0 à rotation continue fonctionne à 0 %
  servomoteur P1 à rotation continue fonctionne à 0 %

toujours
  si Ultrasonic Sensor (in cm) at P8 < 20 alors
    servomoteur P0 à rotation continue fonctionne à 0 %
    servomoteur P1 à rotation continue fonctionne à 0 %
  sinon
    servomoteur P0 à rotation continue fonctionne à 50 %
    servomoteur P1 à rotation continue fonctionne à -50 %
    pause (ms) 1000
  
```

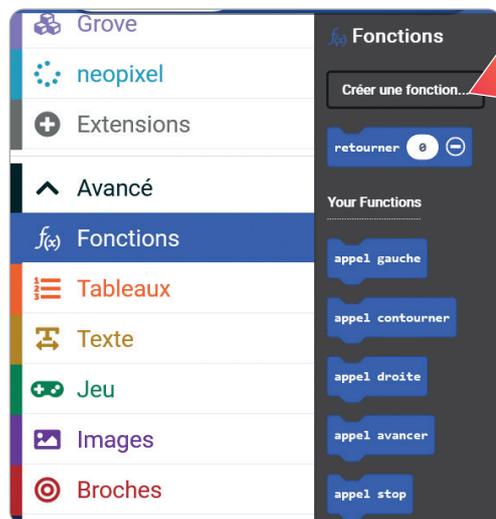
Il ne reste plus qu'à charger une nouvelle fois le fichier modifié sur la carte micro:bit et de regarder comment se déplace ton robot!

5

Contourner un obstacle!

En te basant sur l'activité précédente et les activités de la partie 2, essaie de programmer ton robot afin qu'il contourne un obstacle. Tu peux utiliser une boîte à chaussures, un sac de sport...

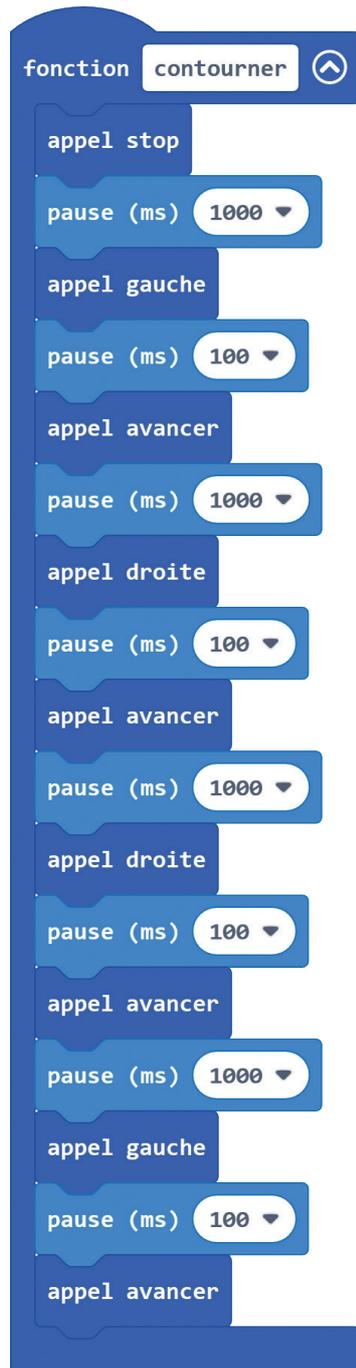
On te donne quelques indices pour te lancer dans ce script. Il faut aller dans les blocs de commande avancés. Choisi « **Fonctions** » et créer une nouvelle fonction.



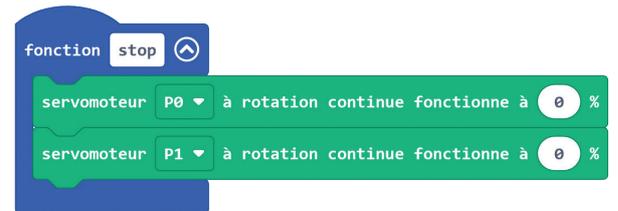
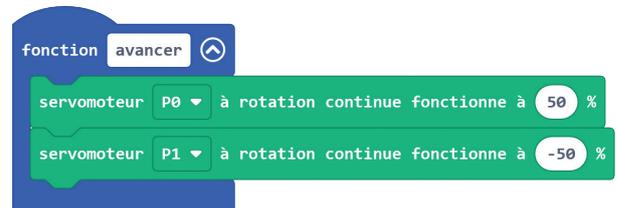
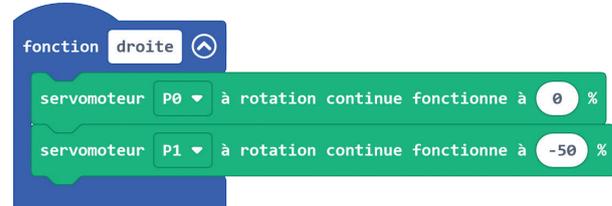
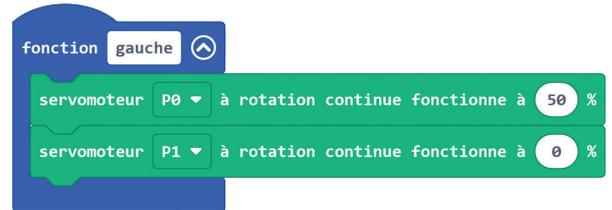
Tu crées alors une nouvelle fonction « **contourner** ».



Et toujours dans le menu « **Fonctions** », avec les différents blocs « **appel** », écrit ce script.



Enfin, tu peux utiliser toujours les blocs « **fonctions** » de cette manière :



Tu es bloqué? Tu peux utiliser le programme exemple: **robot_geek_tuto5_4.hex** et charge-le sur ta carte **micro:bit**. Attention, tu auras tout de même à adapter les valeurs de vitesses et temporisation en fonction de ton parcours.