



Club robotique / SNT

Séance 5 : Déplacement du Cat-Car Evitement d'obstacle

Heures numériques
2024-2025
Robotique éducative

Voici le code du programme « *pgme_avoid obstacle_Janv2025* » .

Objectifs :

Partie 1 : Analyser des morceaux de code,

Partie 2 : Télécharger le programme et enregistrer le dans *code.py* du *CIRCUITPY*, exécuter le code et observer le déplacement du robot.

Partie 3 : Effectuer des modifications sur certains paramètres (vitesse, angle de rotation des roues...)

Partie 1 : Lire les lignes de code en Python ci-dessous et compléter les descriptions correspondantes

Vous pouvez utiliser un logiciel d'intelligence artificielle pour vous aider à décrypter le code et ouvrir d'autres fichiers disponibles avec le robot.

```
1 from BPI_PicoW_S3_Car import Motor, Servo, I2CPCF8574Interface, LCD, LCD1602, ultra
2 import time
3 import board
4 import busio
5 import pwmio
6 import digitalio
7
8 motor = Motor()
9
10 speed_high = 50
11 speed_low = 26
```

Description des lignes 1 à 11 :

```
12
13 def display_distance(lcd_print):
14     lcd_putstr = LCD1602()
15     lcd_putstr.lcd.clear()
16     distance = ultra.get_distance()
17     lcd_putstr.lcd.print(str(distance))
18     lcd_putstr.lcd.print("\n")
19     lcd_putstr.lcd.print(lcd_print)
20     old_lcd_print = lcd_print
21     lcd_putstr.lcd.close()
22     time.sleep(0.3)
```

Description des lignes 12 à 22 :

```

23
24 def avoid_obstacles():
25     lcd_print = " Bonjour!"
26
27     distance = ultra.get_distance()
28     if distance > 60:
29         motor.move(1, "forward", speed_high)
30         lcd_print = "J'avance vite"
31         display_distance(lcd_print)
32
33     elif 20 <= distance <= 60:
34         motor.move(1, "forward", speed_low)
35         lcd_print = "Je ralentis"
36         display_distance(lcd_print)
37
38     elif distance < 20:
39         motor.motor_stop()
40         motor.move(-1, "backward", speed_low)
41         lcd_print = "Obstacle devant"
42         display_distance(lcd_print)
43         motor.move(1, "left", speed_low)
44         lcd_print = "Je tourne"
45         display_distance(lcd_print)
46         motor.motor_stop()
47         distance = ultra.get_distance()

```

Description des lignes 24 à 47:

```

48
49 if __name__ == '__main__':
50     while True:
51         avoid_obstacles()
52

```

Description des lignes 48 à 51:

Partie 2 : Télécharger le programme et enregistrer le dans *code.py* du *CIRCUITPY*, exécuter le code et observer le déplacement du robot.

Décrire le déplacement du robot :

D'après vos observations, qu'est ce qui fonctionne ?

D'après vos observations, qu'est ce qui pourrait être améliorer ?

Partie 3 : Effectuer 2 modifications sur certains paramètres (vitesse, angle de rotation des roues...)

1_ Lignes modifiées : <hr/> Conséquences sur le déplacement du robot : <hr/>	2_ Lignes modifiées : <hr/> Conséquences sur le déplacement du robot : <hr/>
---	---