

# Guide de Programmation Arduino

## Rappels des bases

### Structure d'un programme Arduino

Un programme Arduino (appelé *sketch*) comporte deux fonctions principales :

- `void setup()` : Fonction exécutée une seule fois au démarrage du programme. Elle sert à initialiser les variables, les broches, la communication série, etc.
- `void loop()` : Fonction exécutée en boucle après `setup()`. C'est ici que se trouve le code principal du programme.

### Syntaxe de base

- **Variables** : Permettent de stocker des données (`int`, `float`, `boolean`, etc.).
- **Conditions** : Permettent d'exécuter du code seulement si une certaine condition est vraie (`if`, `else`).
- **Boucles** : Permettent de répéter une section de code (`for`, `while`).

## Fonctions spécifiques

### Lecture du capteur

- `digitalRead(pin)` : Lit l'état haut ou bas d'une broche numérique.
- `analogRead(pin)` : Lit une valeur analogique (entre 0 et 1023) sur une broche analogique.

### Contrôle des moteurs

- `digitalWrite(pin, value)` : Écrit un état haut (`HIGH`) ou bas (`LOW`) sur une broche numérique.
- `analogWrite(pin, value)` : Écrit une valeur PWM (Pulse Width Modulation) entre 0 et 255 sur une broche supportant le PWM.

## Exemples de code utiles

### Implémenter une condition basée sur le capteur

```
1 // Si la distance mesurée est inférieure à un seuil, le robot tourne
2 if (distance < seuil) {
3     // Code pour faire tourner le robot
4 } else {
5     // Code pour faire avancer le robot
6 }
```

### Contrôle des directions du robot

- **Avancer** :

```
1 // Active les moteurs pour avancer
2 digitalWrite(moteurGaucheAvant, HIGH);
3 digitalWrite(moteurGaucheArriere, LOW);
4 digitalWrite(moteurDroitAvant, HIGH);
5 digitalWrite(moteurDroitArriere, LOW);
```

- **Reculer** :

```
1 // Active les moteurs pour reculer
2 digitalWrite(moteurGaucheAvant, LOW);
3 digitalWrite(moteurGaucheArriere, HIGH);
4 digitalWrite(moteurDroitAvant, LOW);
5 digitalWrite(moteurDroitArriere, HIGH);
```

— **Tourner à droite :**

```
1 // Active les moteurs pour tourner a droite
2 digitalWrite(moteurGaucheAvant, HIGH);
3 digitalWrite(moteurGaucheArriere, LOW);
4 digitalWrite(moteurDroitAvant, LOW);
5 digitalWrite(moteurDroitArriere, LOW);
```

— **Tourner à gauche :**

```
1 // Active les moteurs pour tourner a gauche
2 digitalWrite(moteurGaucheAvant, LOW);
3 digitalWrite(moteurGaucheArriere, LOW);
4 digitalWrite(moteurDroitAvant, HIGH);
5 digitalWrite(moteurDroitArriere, LOW);
```

## Bonnes pratiques

- **Commenter votre code** : Ajoutez des commentaires pour expliquer les parties clés de votre programme.
- **Utiliser des noms de variables explicites** : Facilite la lecture et la compréhension du code.
- **Structurer le code** : Séparez les différentes parties du code avec des sauts de ligne et des commentaires.

## Dépannage courant

- **Problèmes de compilation** : Vérifiez la syntaxe, les parenthèses et les points-virgules.
- **Problèmes de connexion** : Assurez-vous que les broches sont correctement connectées et que les numéros de broches dans le code correspondent au montage.
- **Problèmes logiques** : Utilisez la fonction `Serial.print()` pour afficher des valeurs et comprendre le comportement du programme.

## Ressources supplémentaires

- Référence Arduino (français)
- Tutoriels Arduino en français
- Forum Arduino pour poser des questions