**Travail pratique 3 - Contrôle d'un robot**

**420-22P-SI**

**Hiver 2023**

**Informations préliminaires**

- Date de remise de la description du travail : Lundi le 24 avril 2023.

- Date de remise du travail complété : Jeudi le 11 mai 2023, avant 17h00.

- Représente 50% de la note finale.

- Travail personnel.

- Environnement de programmation : Processing, Arduino et Tinkercad.

- Vous devez remettre sur **Léa** le dossier de votre projet, sous l'énoncé **Travail pratique 3**.

**Description générale**

Effectuez, à l'aide d'une interface graphique en Processing, le contrôle des mouvements du robot mobile **Cherokey 4WD** de DFRobot.

Les mouvements du robot sont contrôlés à l'aide de ses 4 roues motorisées.

Sur le robot est embarqué une carte **Arduino Uno WiFi**, laquelle contrôle les mouvements du robot.

Le lien avec le robot est effectué via une connexion Wi-Fi.

Votre interface graphique doit permettre de contrôler les mouvements suivants :

- Avancer en ligne droite.

- Reculer en ligne droite.

- Arrêter.

- Pivoter sur la droite.

- Pivoter sur la gauche.

- Avancer en tournant vers la droite, avec une intensité variable.

- Avancer en tournant vers la gauche, avec une intensité variable.

- Reculer en tournant vers la droite, avec une intensité variable.

- Reculer en tournant vers la gauche, avec une intensité variable.

Notons que les 5 premiers mouvements ci-dessus sont effectués avec des moteurs à pleine vitesse.

L'intensité d'un virage peut aller de 0 (aucun virage) à 5 (virage serré avec seulement les roues d'un seul côté qui tournent), avec un incrément de 1.

Le protocole à respecter entre Processing et Arduino est composé de deux octets, le premier étant le mouvement à réaliser, le second l'intensité des virages.

Le contrôle de votre robot devrait être possible à partir de plusieurs clients connectés.

Afin de sauver du temps et économiser les piles du robot, veuillez d'abord réaliser votre système de contrôle en simulant les mouvements du robot. Lorsque cette simulation saura répondre aux exigences du système de contrôle, des modifications mineures pourront être effectuées afin que les mouvements du robots physique puissent être contrôlés via le réseau.

**Étape 1 - Tinkercad et le robot virtuel**

Utilisez la modélisation du robot sur Tinkercad. Simulez l'envoi et la réception d'octets de contrôle (selon le protocole) en utilisant le Moniteur série (voir la simulation **Simulation d'une réception d'octets**).

**Étape 2 - Processing et l'interface de contrôle**

Créez votre interface de contrôle. Simulez l'envoi d'octets de contrôle (selon le protocole) en affichant ces derniers dans la console d'affichage.

Du temps peut être consacré à l'extérieur des heures du cours pour réaliser les deux étapes précédentes. Ces dernières doivent être complétées avant d'avoir accès au kit Arduino.

**Étape 3 - Processing et Arduino**

La programmation réseau est ajoutée. Simulez les mouvements du robot et l'intensité des virages à l'aide de LEDs, tel qu'illustré ci-dessous.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Avancer** | **Reculer** | **Arrêter** | **Pivoter sur la droite** | **Pivoter sur la gauche** |
|  |  |  |  |  |
| **Tourner à droite en marche avant** | **Tourner à gauche en marche avant** | **Tourner à droite en marche arrière** | **Tourner à gauche en marche arrière** |  |
|  |  |  |  |  |

L'étape précédente doit être complétée avant d'avoir accès au robot physique.

**Étape 4 - Processing, Arduino et le robot physique**

Modifiez votre code Arduino afin qu'il puisse contrôler les mouvements du robot physique.

**Étape 5 - Ajout d'une LED au robot physique**

Afin que le robot puisse indiquer qu'il est connecté au réseau Wi-Fi, une LED doit être allumée aussitôt la connexion réalisée. Si cette connexion est perdue, la LED doit être éteinte avant que le robot ne tente de la rétablir.

**Étape 6 - Transmission de données Access**

Sauvegardez le code à chacune des étapes précédentes. Si vous ne parvenez pas à réaliser toutes les étapes, la dernière complétée servira pour la correction.

Vous devez démontrer en classe le fonctionnement de votre système de contrôle, avant la date limite de remise du travail.

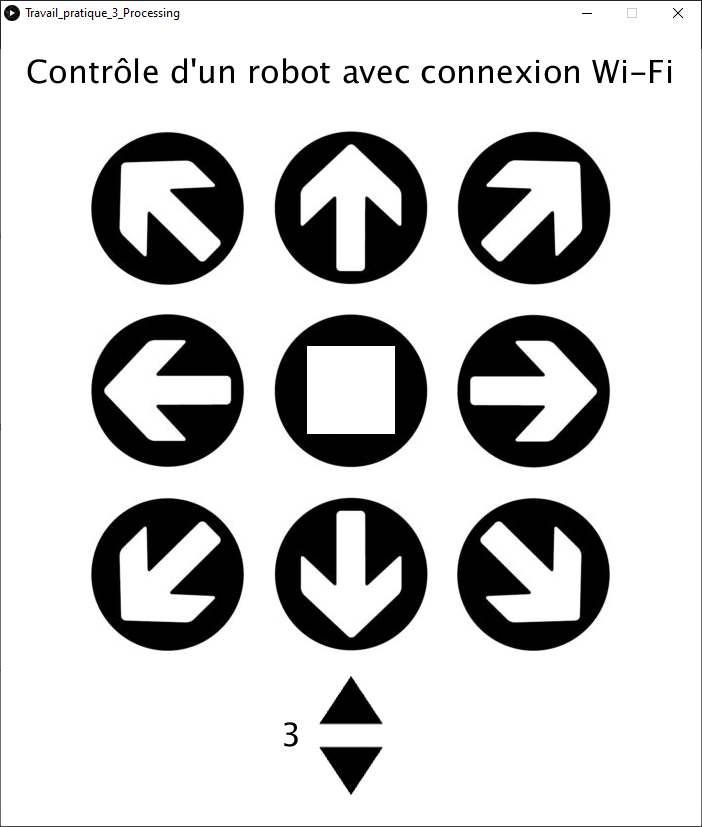
Vous n'avez pas à remettre d'algorithmes. Toutefois, il est **fortement** recommandé d'en produire avant de vous lancer dans la programmation, et ainsi gagner du temps.

Vos programmes en Processing et en Arduino doivent posséder un en-tête complet, et être bien commenté aux endroits appropriés.

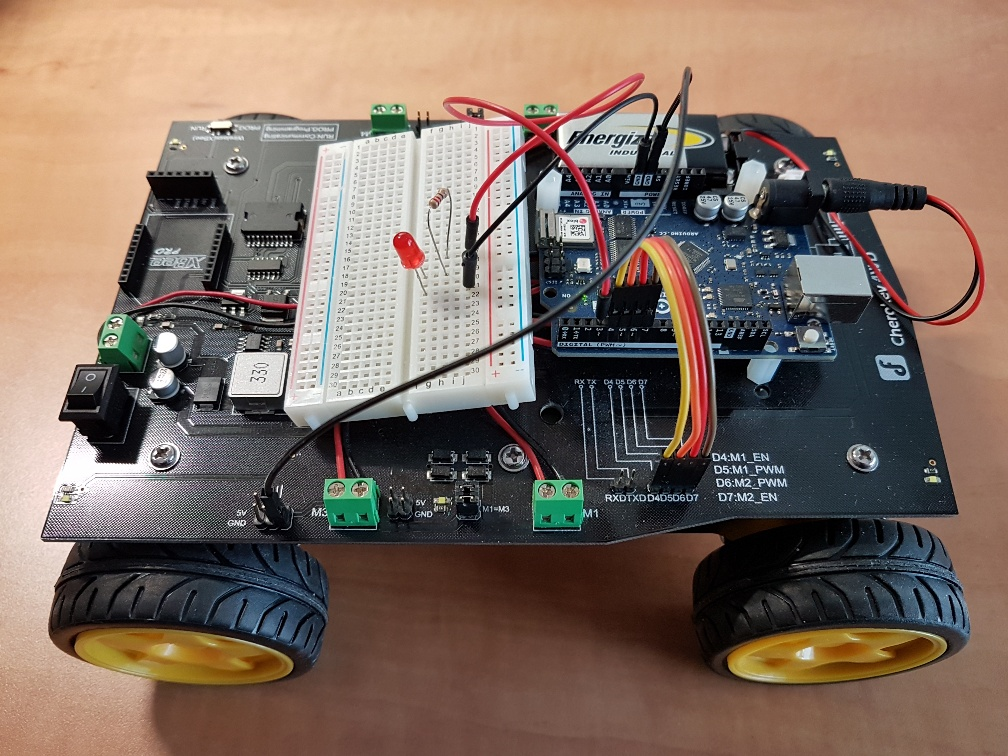
Vos variables doivent posséder des noms significatifs et être commentées.

La présentation de votre code doit être impeccable au niveau de sa modularité, de sa structure et de son indentation.

**Interface Processing**



**Configuration physique du robot**



**Directives importantes**

- Débranchez l'alimentation 9V avant de brancher le câble USB.

- Débranchez le câble USB avant de rebranchez l'alimentation 9V.

- Mettre l'interrupteur du robot à ON seulement lorsque le robot est au sol.

- Reliez le GND du robot avec le GND de la carte Arduino.

**Autres informations**

N'hésitez pas à utiliser la console de Processing pour le dépannage, avec les fonctions **print()** et **println()**.

N'hésitez pas également à utiliser le Moniteur série d'Arduino pour le dépannage, avec les fonctions **Serial.print()** et **Serial.println()**.