Introduction to Robotics: Homework

Andrei Dumitriu andrei.dumitriu@fmi.unibuc.ro

Last Updated: October 21th 2022

Latest homework: Homework #0

1 Description

General guidelines for the document and all homework:

- Each homework is on a different page, despite the remaining page space.

- You must come with the homework already implemented, both hardware and software (food casserole is a great option for carrying your project). Installing or uploading the code at the lab leads to deduction in points.

- Code must already be uploaded to Git. We will only take into account the last upload before the deadline, not before the presentation.

- Coding style is of utmost importance in this lab.

- Color coding is of utmost importance in this lab.

- Some homework might have hard and soft deadlines, but **only if explicitly specified**.

The homework's purpose is to help you learn and practice the knowledge gained in this lab. We place strict emphasis on **correctly implementing** the requirements, not just getting it done. If you struggle with it, do not hesitate to contact your Teaching Assistant or Andrei directly in Microsoft teams.

2   Homework #0

Deadline (hard): Sunday, 23rd of October, 23:59:59.
The first homework is easy, and consists of two parts:

1. Install arduino IDE: `https://www.arduino.cc/en/software`

2. Create a **public** Git repository , according to these specifications:

   - You can use this as an example (will be updated as we homework progresses): `https://github.com/Irikos/IntroductionToRobotics`

   - Repository must be name "IntroductionToRobotics" (or something very similar, but with a good reason)

   - Must include similar Readme.md with title, initial description and subsection for each homework (with specified details)

   - You are free (and encouraged) to be creative with your repository description. **However**, the freedom of creativity must not be confused with and excuse to be lazy. The requirements specified here and in the repository description are considered basic and must be met

   - You can use other options, like Gitlab, Bitbucket etc as long as it meets the same basic requirements

   - Use the "Turn in" button in the teams assingment when done. Include the git link (make sure it's public!) in the text box. DO NOT upload a file with the link.

These must be completed until Sunday, 23rd of October, 23:59:59.

3   Homework #1

Deadline (hard): Your respective lab in the week of October 24-30.
As was described in the laboratory, the homework consists of the following:

- **Components**: RBG LED (1 minimum), potentiometers (3 minimum), resistors and wires (per logic)

- **Technical Task:** Use a separat potentiometer in controlling each of the color of the RGB led (**R**ed, **G**reen and **B**lue). The control must be done with **digital electronics** (aka you must read the value of the potentiometer with Arduino, and write a mapped value to each of the pins connected to the led.

- **Publishing task:** You must add the code to the Git repo and continue updating the readme with at least the following details (but feel free to be more creative):

  1. Task Requirements
  2. Picture of the setup
  3. Link to video showcasing functionality (I recommend youtube, but anything I can access is fine)
  4. Remember to publish the video in the correct orientation. Don't do this: `https://youtu.be/Y8HOPlUtcto`
  5. **Hand in the homework on MS teams when done - aka when git is up to date**

- **Coding task:** Coding style is of utmost importance. You must have a perfectly clean code in order to receive the maximum grade. Keep in mind that magic numbers are not accepted, although you might see them in the lab (as a trade-off for speed). Remember to be consistent in your style, check the style guide and the provided style document and use correct spacing.

  Example used in lab which would not yield the maximum grade: "analogWrite(ledPin, potValue / 4) or "voltage = potValue * 5.0 / 1023.0".

  Mistakes observed there:

  1. "potValue / 4" is not precise, since the value does not always split into 4 perfectly. Use the **map() function**
  2. We calculate the value inside the analogWrite function
  3. **potValue** and **potPin** are not ideal naming versions. What happens if you change your potentiomer with a slider? (a slider has the same output range, so no change to the code would be needed - if written correctly)
  4. **4** is a magic number. **5.0** and **1023.0** are magic numbers. They should be replaced with constants, such as **maxAnalogValue**, **maxVoltageValue** etc