University of Tehran
College of Engineering
School of Electrical and
Computer(ECE)
School of Mechanical
Engineering(ME)

Mechatronics & Robotics

Project 3: Robot Operating System(ROS)

Teaching Assistants:
Elnaz Balazadeh

Deadline: 4 June 2024 (15 Khordad), 23:59

Table Of Contents

Part 1: Draw character with turtles! (20 points)

In this project, I want you to draw some characters with turtlesim turtles. As you know turtlesim is the 2D simulator of ROS which gives you plenty of options to implement your ideas or develop current algorithms. In this simulator, there is a controllable turtle that is the main element of this project.
For each of you, two characters have been specified which are available in the attached Excel file. For this part of your project use as many turtles as you need to form the specified characters. A sample is provided in the Figure 1.

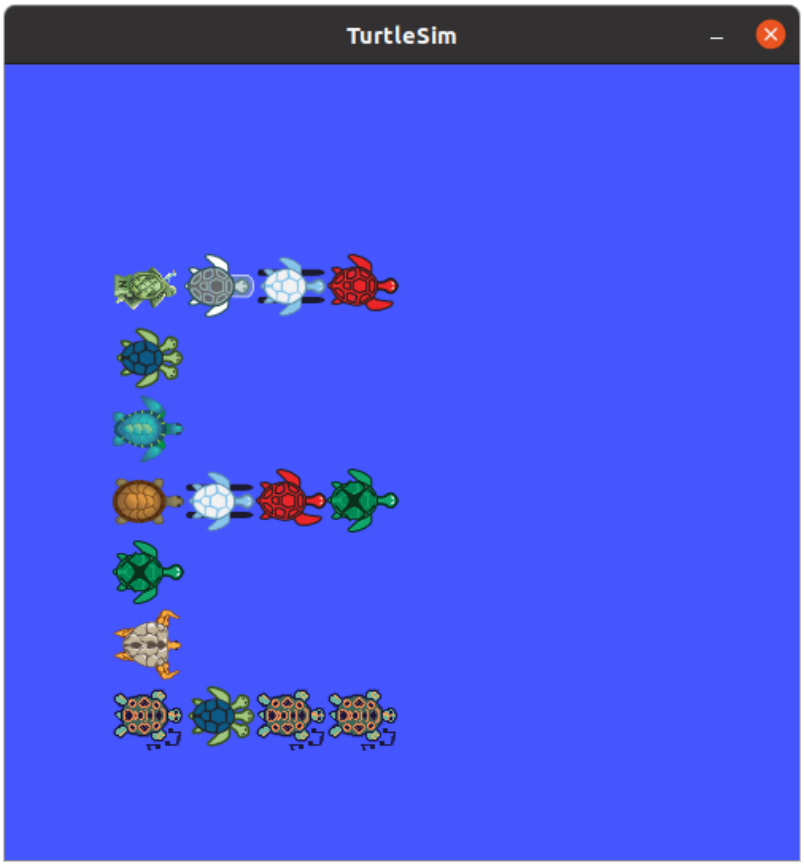

Figure 1: A sample of writing letter **E** with turtles in turtlesim.

Part 2: Rotating the turtles! (15 points)

Well, in this part, let's do some exciting performance. As mentioned before, your turtle is controllable and you can rotate or move it by sending the proper topic. Use the `rostopic list` and explore related topics for linear and angular velocity. Write a Python script that rotates all turtles on the screen at the same time. Name this node as "rotating_node". Explain your approach and name all topics and types of messages that you used for this part.

Bonus (20 points)

Write a Python script that makes it possible to use the arrow keys to move the first displayed letter in the turtlesim playground.

Part 3: Launch files and Report (65 points)

Note that this part is the most important section of your project and if it doesn't have a launch file, YOUR PROJECT WILL NOT BE SCORED. So, to run your project, write a launch file that contains all required nodes for the complete operation of drawing letters. By running this launch file, the turtlesim_node should be executed and your assigned letters appear. After that, all turtles should start rotating till `ctrl` + `c` stops the process.
Beside this launch file, write a summary of what you did to solve the challenges that you faced during this project. Your report should contain a complete explanation of all message types, services, and topics that you used and also plot the graph of all nodes. It is necessary to have an "Error" section in your report. Make sure you explain all the problems you had and how you fixed them in your report. This will help you get a better score. Don't forget to talk about the mistakes you made. And make sure to give clear instructions on how you solved each problem.

Homework Guidelines and Instructions

- The deadline for sending this project will be until the end of Tuesday, June 4th.
- This time cannot be extended and you can use time grace if needed.
- The implementation must be in Python programming language and your codes must be executable and uploaded along with the report.
- This project is done by one person.
- If any similarity is observed in the work report or implementation codes, this will be considered as fraud for the parties.
- Using ready-made codes without mentioning the source and without changing them will constitute cheating and your practice score will be considered zero.
- If you do not follow the format of the work report, you will not be awarded the grade of the report.
- Handwritten project delivery is not acceptable.
- All pictures and tables used in the work report must have captions and numbers.
- A large part of your grade is related to the work report and problem solving process.
- Please upload the report, code file and other required attachments in the following format in the system: `MP3_[Lastname]_[StudentNumber].zip`
  For example, the: `MP3_Balazadeh_12345678.zip`
- If you have questions or doubts, you can contact the assistants through the following e-mail with the subject **MP3_Q**. Stay in touch educationally:
  - ROS Project(MP3): `balazadeh.elnaz@gmail.com` (Elnaz Balazadeh)
- Be happy and healthy