Laboratory exercise 4

ROS: Introduction

Name: Luka Habuš **JMBAG:**

Preparation

- Review the ROS lecture slides.
- Make sure you are familiar with the *turtlesim* package.
- If you haven't done so already, create a catkin workspace.
- Install *pynput*, python library for controlling and monitoring input devices, by running:

```
pip3 install pynput
```

Assignments

Task 1: Mouse Tracker

a) Clone the prepared ROS package, *mouse_tracker*, in your catkin workspace, start *roscore* and run *mouse_position_publisher.py* node from the cloned package. Which command did you use to run the node?

rosrun mouse_tracker mouse_position_publisher.py

b) Once you started the node, go ahead and check which topics does it publish. Which command are you using for listing ROS topics?

rostopic list

c) As you could have noticed, *mouse_position_publisher.py* node publishes only one topic, called */mouse_position*, on which it publishes the current position of your mouse. In the first text box, write the command which enables you to check the type of the message published on the */mouse_position* topic and in the second text box write the obtained message type.

rostopic info /mouse_position

Type: geometry_msgs/Point

d) To verify that the *mouse_position_publisher.py* node runs as it should, print the published messages directly from the terminal. Write the command you have used for printing the messages.

rostopic echo /mouse_position

e) Inspect the coordinates of the upper left and bottom right corner of your screen. What is your screen resolution?

1920 x 1080

f) Inside the *mouse_tracker* package create a new folder called *launch*. Write a launch file, *track_mouse.launch* which will run *mouse_position_publisher.py* node and *turtlesim_node* from the *turtlesim* package.

Exercise submission

Create a zip archive containing this pdf file with the filled out answers and the *mouse_tracker* package. Upload the archive to Moodle.