

CS 69.13, Fall 2022, Elliot Potter

Method Description

This program is a modified version of the `simple_motion` program that you provided to us. I decided to import that program as a library, and then write a separate class to handle subscribing to odometry and publishing error messages. My program essentially is a for-loop where the robot drives forward, then rotates by $360 \text{ deg} / \text{num_sides}$. It accepts parameters from the launch file, and these determine the size of polygon, number of sides, and direction of travel.

The package is structured around a single node, which implements both the `simple_shapes` class as well as the driver function. I chose not to separate the class from the driver because placing the class in the `./src/simple_shapes` folder meant that it could not access the `simple_motion.py` module. There is probably a solution to this, but I'm leaving that as a todo. The package also has three launch files for different shapes, which I decided was a more intuitive way of drawing shapes than passing in command-line arguments.

Evaluation

This program worked fine. I essentially wrote it without testing it, and it pretty much worked as soon as I was able to get the file to run. The only unexpected issue was issues with the robot driving clockwise and counterclockwise – I had to reverse the robot's rotation direction if it drove clockwise.

I had the most difficulty figuring out the configuration of the package – I ended up iteratively converting `simple_motion` into the final package so that I knew exactly what was preventing the build. I ended up with a set of rules around creating ROS packages (not included in this zip).