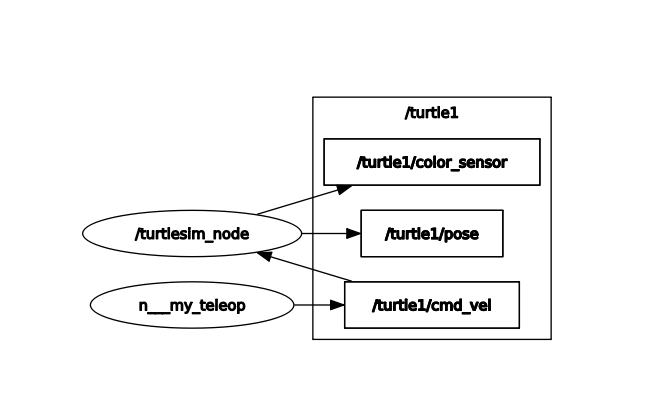
Task 1 is about teleoperation using the keyboard, with an option to change the speed:

In this task I used “getch” to capture keyboard inputs and will pass it to the turtlesim bot, movement will occur when using the “Up, Down, Right and Left key arrows on your keyboard”. The bot here will move along x and y axis while also rotating along the z axis. I was not able to implement the option to decrease and increase the speed using the keyboard.

RQT Graph for task 1:



Task 2 is to navigate to a given coordinate:

For this x and y coordinates will be inputted to the turtlesim bot and the turtle will be able to navigate to it by calculating the linear and angular velocities.

I faced and issue here where if am using close coordinates it will spin out, this was fixed by removing the constant set speed in my code.

RQT Graph for task 2:

A diagram of a computer program

Description automatically generated

Task 4 which is a Roomba like behavior for 1:

Using math library, I was able to turn in exact angles (pi/s) which helped me a lot instead of having a random turn value. My bot will turn and move forward, stop and check surroundings then move again accordingly.

RQT Graph for Task 4:

A diagram of a software developer

Description automatically generated

Task 5, like task 4 will have 5 turtlesim bots exhibiting a Roomba like behavior:

This was possible by utilizing the spawn service and implementing it in my code to call upon 5 bots with preset locations and angles. I choose for them to spawn facing away from each other for neater start.

I am facing some issue here when it comes to wall and other turtle detection, if it gets close to two turtles then they might hit depending on angle of entry.

However, I have seen some split the window into 4 grids and spawning 4 turtles that have their set boundaries and will mimic the same program for one. This will run simultaneously.

I was not able to do that due to time constrains and me wanting to spawn more robots, I should have considered the fact that overcrowding might impact the neatness of the running program.

RQT Graph for Task 5:

A diagram of a flowchart

Description automatically generated