

## Self eval tasks

1. Run turtlesim node. Send a command to run the turtlesim backwards at 5Hz rate.
2. Create a ros package "timer package" with dependencies rospy. Add a dependency std\_msgs to the package. Make a rosnod that publishes current time to the console every 0.5 secs.
3. Write a program to move turtle 1m forward from its current position, rotate it 45 degrees and stop. Print the initial and final position of turtle.
4. Install turtlebot3. Launch turtlebot3\_gazebo turtlebot3\_world.launch and inspect the topics, nodes, rqt\_graph

## Eval Tasks:

Task 1: Make two Ros nodes. Node 1 will publish a string taken as input from the user and Node 2 will subscribe to it and will print it out on another terminal. Use the Ros wiki and beginner tutorials for assistance.

Task 2: Make a service server and client in Ros. The client must send two floats and an integer to the server. The server shall take the two floats as operands and perform the operation as defined by the integer and return the result (float) and a Boolean. If there were no errors, the float is set to the result and the Boolean is true, else the float is set to zero and the Boolean is false. This result is then printed out with the Boolean value.

Integer-

1+

2-

3\*

4/

Task 3: Make an action server for taking feedback from user until he types a given sequence A. You will make two nodes in this - client and server. On starting the client, it will take input from the user for the value of A. A is then sent to the server. The server then keeps taking input from the user until A is entered. The feedback provided is the length of the input provided by the user. The result is a composite message of the number of feedback messages sent and the time elapsed from sending of the request to the goal message.