**Assignment 2 - Turtlesim Movement: Rectangle**

**Due: 3/06/17 Time: 11:55pm**

**a) Team Name: Cyberdyne Industries**

Stephen Chan, [stephen.chan9@yahoo.com](mailto:stephen.chan9@yahoo.com) (25% contribution)

Reyniel Maglian, [renrenmaglian@gmail.com](mailto:renrenmaglian@gmail.com) (25% contribution)

Vincent Campbell, vrcampbell@csu.fullerton.edu (25% contribution)

Mohammad Hassnain, [mohammed-17@csu.fullerton.edu](mailto:mohammed-17@csu.fullerton.edu) (25% contribution)

**b)**

**i) Description of the strategy used to accomplish our goal:**

Using the framework created by the turtlesim robot\_cleaner.cpp we adjusted the parameters of the functions move() and rotate() in order to match the directions of the edges to form a rectangle. The turtle had the same movement and distance for opposite sides and would later adjust to the shorter side of the rectangle. Rotation speed stays primarily the same. Basically, our strategy was to make sure the turtle moved and rotated in the shape of a rectangle. We considered all the other options of rotating and moving the turtle, and ultimately decided on the best movements based on what kind of shape we want the turtle to move in. In this case, we decided on constant values for each side of the rectangle. We decided to make the speed and rotation degree constant being 7 for speed and 90 degrees for rotation. For the rotation we constructed it so it would rotate clockwise. The distance is what was not constant, because we wanted the turtle to move in a rectangle. We had the length distance set to 4 and the width distance set to 2. After implementing this all, we got it to work and move in a rectangle.

**ii) Psuedocode for robot\_cleaner.cpp:**

-The beginning of the code basically initializes all the variables we need to use and implement for the code. We have to initiate the new ROS node name “talker”

-After all the declarations and initializations, the program moves on in main and starts the program first by outputting “ROS Turtle Rectangle Project”

-The program then calls the function move() with the speed, distance, and moving forward or backward as the parameters.

-The function rotate() is then called with the degrees of how far to rotate the turtle, and which way to rotate it (clockwise or counter-clockwise) as its parameters. This program was coded to rotate it clockwise.

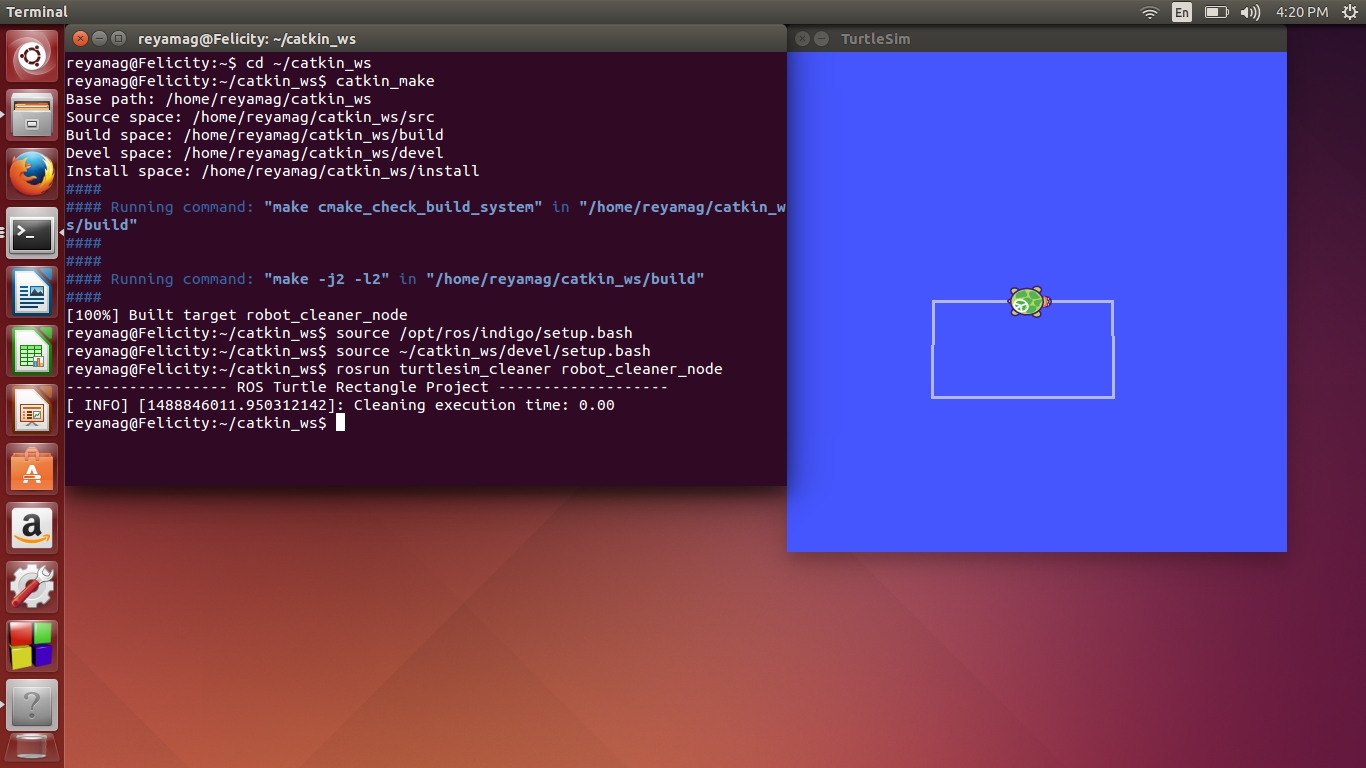
-Within the rotate function, it called the degrees2radians() function to convert the

entered desired degrees to radians.

-After the program executes this far, it then calculates the cleaning execution time.

-The program then ends and the turtle has successfully moved autonomously in a rectangle.

**c) Program result screenshot:**



**d) Resources:**

**Resources Used:**

* Edu.gaitech.hk
* <https://github.com/aniskoubaa/gaitech_edu/blob/master/src/turtlesim/cleaning_app/robot_cleaner.cpp> (code from the video)

**YouTube resources:**

ROS Tutorial 4.2: Moving in a Straight Line (Turtlesim Cleaner)

Covers robot\_cleaner.cpp setup and execution

<https://www.youtube.com/watch?v=PGZMlzBlMmw>

ROS Tutorial 4.3: Rotation Left/Right (Turtlesim Cleaner)

<https://www.youtube.com/watch?v=Ddqwq2WXFEk>

ROS Tutorial 4.4: Go-To-Goal Location (Turtlesim Cleaner)

<https://www.youtube.com/watch?v=Qh15Nol5htM>