Jeff Khvu, Raymundo Valadez, Yathim Pang

Group 5

Project 1: Tracing the Shapes

In this project we were given the task of programming our robot to travel along designated paths by using NXTRegulatedMotor commands as well as the DifferentialPilot class provided to us. We had to create objects for our robot to use in its main classes so we created a separate class known as MyRobot to hold the methods we could use in the main classes. We utilized the rotate method on each of our motors to help our robot move forward and turn. We would have the first rotate method hold a Boolean to allow the next line to run at the same time, thus allowing us to move forward and not spin in circles. When we did need to rotate our robot, we used rotate in opposite directions for each motor to allow our robot to turn the corners of the square.

For the second and third milestones, we used the DifferentialPilot class to successfully drive forward and rotate in the designated shapes. We had to convert the measurements of our tires and the distance between tires to centimeters in order for us to keep our values consistent. We used the travel and rotate methods in the DifferentialPilot constructor to complete the second milestone by having the travel method move the robot forward and then rotate to turn the robot. To complete milestone three, we utilized DifferentialPilot as an argument for a PolyTester class we created. We also created a separate PolyTracer class which we used to create an object within PolyTester. We used the travel method to first travel along the straight portions of the shapes and then the rotate method to achieve the angle needed for each angle of the shape. The rotate method had to take the supplement of the interior angle of the shape to get the correct rotation. For the semicircle and quarter circle, we used the arc method to have the robot travel along a 36 inch radius semicircle and a 12 inch radius quarter circle. We had to adjust the radius and degrees slightly to correct the errors when traveling along a long path.

The most difficult part of the project was just doing many trials to change values and adjust our robot to achieve the necessary milestones. Constantly changing the track width by fractions of a centimeter and then testing the robot became very tedious. The most interesting part was getting the robot to perform the task and return close to its starting position. When we did our trials, we ended up getting about one to two degrees of error in angle and about two to three inches of error for distance. We found that sources of variability came from the ability of motors to stop at exactly the correct rotation. When we applied the tacho count, the readings were either one or two counts too many or too few. Another source of variability came from the condition of our tires. The cleaner our tires were, the better our robot would perform, and when they were dirty our errors seemed to be greater.

Files used:

SquareDance.java – holds main method of Milestone1

DifPilot.java – holds Milestone2 calibrating the DifferentialPilot

SquareTracer .java – holds main method of MileStone 2

PolyTester – Holds main method of Milestone 3

GenericRobot – holds methods to call robot to move. Methods are used in all Milestones.