

COMP20170 Introduction to Robotics Challenge 1

Group Journal

24/01/20

* Initially we had issues installing the firmware to the robot because conflicting programs were running. John eventually figured out the solution and resolved it the issue.
* John, Peter and Matthew built Derek (we named it). Conor and Finbar worked on installing firmware and software on the laptops we were using.
* ROBOTC was also installed on a Windows virtual machine running in Linux.
* When assembly: we connected two motors to EV3 and wheels, with a ball bearing on the front.
* Next, we attempted different variants of the sample code and worked out that turning the robot based on speed of motors (50 -50) rather than sleep time.
* We experimented with loading in custom sound effects etc.
* A while loop was used to repeat behaviour.
* We played around with attaching an arm to the robot to swing at the end of its movements to give Derek offensive capabilities.
* Conor took notes and logs of the progress of the robot to that point.

31/01/20

* After assembling Derek last week, we added the more important parts today such as the sonar sensor, the touch sensor and the colour sensor.
* Daniel, John and Finbar went through the available Lego parts and collectively thought of the best ways to tackle attaching each sensor.
* Conor worked on adjusting and perfecting the code.
* It was important for every sensor to be in the best possible position on the robot to avoid any errors from physical obstruction of sensors. For example, we attached the sonar sensor first and we ran into an issue with the placement of the colour sensor.
* The touch sensor had to be beyond the sonar sensor, and it had to be close enough to the body to not compromise the reaction time of the robot. We found that any way we tried to attach it; the touch sensor would obstruct the sonar sensor as a result.
* We thought about solutions to resolve this issue and decided that extending arms out, slightly further than the sonar sensor, would be a better way to attach the touch sensor while also not affecting the reaction time of the robot’s time.
* Setting up a mini trail on our table using the same red tape, we tested out the code that had been written.
* The robot followed the red line very well on the first try but then gave up towards the end and nearly fell off the table.
* By tweaking the code, we adjusted the robot to make sure it followed the red tape for the full length.
* Derek wobbled a lot on the second attempt, so we thought the code that determined his rotations were probably needed to be reduced.
* Daniel took down notes and logs of what the robot was doing and what was needed to improve and then wrote more detailed notes.

4/02/20

* John, Peter, Finbar and Conor met up at 9am and tried lots and lots of different things and nothing worked for the first while.
* We attempted to use an arc as a work around for the issue of Derek not following the red tape properly
* Once it would go off the line it would turn in an arc that kept getting bigger until it found the red line, it turned out to be very inefficient as it would’ve taken a very long time for it to successfully correct itself.
* Multiple loops with a break + without
* Adjusting position of sensor
* Daniel joined the group and we put Derek on the course and tried time after time again to get him to follow the line in its entirety
* Daniel took notes and logs of Derek’s progress
* Derek refuses to follow the red line after the second bend.
* Derek would follow the black lines without issue but not the red line.
* We changed the value of the offset and changed the integers to doubles for more accurate tweaking.
* Conor removed a conditional statement and Derek finally started following the red line after the bend.
* Peter implemented the code for the last part of the obstacle
* We downloaded the code onto Derek and tried it, Derek would move successfully until it hit into a wall and then stopped.