COMP20170 Introduction to Robotics Challenge 1

Group Journal

Link for tutorial robot and code is based off of
http://robotsquare.com/2014/07/01/tutorial-ev3-self-balancing-robot/

Tasks:

- Build a balancing Robot from scratch
- Write and Debug code for making the robot balance
- Make the robot balance for 7 seconds
- Make the robot turn right for 7 seconds
- Make the robot turn for 7 seconds
- Log each attempt and any progress made with the robot
- Record any near successful attempts for logs
- Write logs into a group journal

Plans:

These are the plans for each task to be delegated to these team members

Daniel: Will initially help build the robot step by step. Will collect all logs at the end and expand into a more articulate learning journal of brief descriptions of attempts and progress.

John: Will oversee the building of the robot while starting the skeleton code with each section laid out to be further implemented.

Finbar: Will take the skeleton code and implement rough attempts until further progress is made and the code will be improved on

Peter: Will initially help build the robot step by step.

Conor: Will take the roughly written code and clean up and be helped in implementing a cleaner version with more logically written and compartmentalised code.

Matthew: Will oversee the physical tests of the robot and record any semi-successful runs.

Everyone will also take turns keeping the logs for each session which will eventually be collected and made into a more articulate group learning journal of different perspectives.

Logs (edited to be more descriptive)

14/02/20 – *Logs taken by Daniel*

- Peter & Daniel had the task of initially building the robot. We read the online tutorial on how to create the robot and connected parts
- John helped us find parts and sifted through the box when we needed very specific parts for important connections
- We ran into some bumps when we couldn't find certain parts but in the end the pictures online had different colours to the parts we had.
- We managed to fully assemble the robot
- During this, John, Conor and Finbar began planning out what the code should do and how to go about doing it
- John broke down the most basic sections that would definitely have to be implemented into the code.

- Finbar and Conor brainstormed how these sections code be written logically in ROBOTC and asked a TA whenever they needed some further help in the implementation
- Matthew came up with a name for the robot: Noddy. We thought it fitted the robot nicely.
- Daniel took logs of the events of the first implementations of code, assembly and birth of Noddy.

17/02/20 – *Logs taken by Peter*

- Today, we hit some milestones in regards to the code.
- For reference: we used Lauren's EV3 balanc3r code which was linked to us in the assignment brief.
- John, Conor and Finbar used similar function names, variables, inputs and outputs but modified overall to use as an early basis for the code.

```
float getGyroBias(){
        float gyroBias = 0.0;
  int readings = 100;

  eraseDisplay();
  displayCenteredBigTextLine(2,"Calibrating...");
        resetGyro( gyroSensor );
  sleep(1000);

        for(int i = 0; i < 100; i++){
                gyroBias += getGyroRate(gyroSensor);
    sleep(5);
        }
        gyroBias = gyroBias / 100; //get average
        return  gyroBias;

}
```

- The first section was able to be somewhat successfully implemented.

- The first section was to let Noddy balance for around 7 seconds. Matthew and Daniel oversaw this when the code was downloaded to Noddy.
- Noddy balanced for 2-4 seconds before almost falling off the table.
- Further brainstorming followed by the group together.

- We all talked about what tweaking to the code needed to happen to have a more successful run in future.
- Peter took logs on the happenings of the session about the progress in our code and first trial run.

19/02/20 – *Logs taken by Finbar*

- Some adjustments were made to the build of Noddy by Daniel and Peter to reduce rattle that was affecting how well Noddy could balance.
- Matthew charged Noddy while the newest additions to the code were being finished.

- Daniel and John gave some insight into the functionality of the code and Finbar and Conor implemented it.
- Peter's laptop was used to download the code to Noddy as he has a windows machine unlike most who are running Ubuntu on their machines.
- Noddy took another test run and this time was more successful than our last run.
- Noddy was able to balance for longer now but the code implemented for it's first turn wasn't working.
- Good progress was made in regards to balancing.
- Finbar took rough logs and notes of today's progress to be later re-written by Daniel into the journal.

21/02/20 – *Logs taken by Matthew*

- Conor and Finbar revised the entire programme that was written and decided there was a better way to tackle the issues.
- The values for kd and kp were adjusted the most over a course of 3 painstaking hours

```
task main()
{
  float gyroRateBias = getGyroBias();
        float kp = 14; // Constant for proportional controllor
        float ki = 2;
        float kd = 1;
        float offset = 0;
        float tp = 0;
        float integral = 0;
        float lasterror = 0;
        float derivative = 0;
```

- They decided to write another version and decide which one would work better
- Function names and logic was similar but it was easier to debug moving in smaller steps
- This code was downloaded to Noddy and he had better luck
- Matthew recorded the session and Daniel and John oversaw, taking any inconsistencies into account.
- Matthew also logged notes about the further progress that had been made.
- Daniel started to re-write the notes into a more articulate manner into the group journal.

25/02/20 – *Logs taken by John*

- The code for the next turn was implemented and added to the main function.
- The new revised code worked a lot better
- Daniel and Matthew helped with downloading the new practice code onto Noddy

- Matthew recorded this attempt
- John logged the events and happenings
- Daniel further wrote the logs into the learning journal

1/03/20 – *Logs taken by Conor*

- New improved code was implemented and we completed our final trial sessions
- Peter and John helped correct the code and troubleshoot the newest issues that had arisen.

```
while(!getButtonPress(buttonEnter)){

        float gyrovalue = getGyroDegrees(S2);
        float error = gyrovalue - offset;
        integral = integral + error;
        derivative = error - lasterror;
        float Tilt = kp * error + ki*integral + kd*derivative;
        float powerA = tp + Tilt;
        setMotorSpeed(motorA,powerA);
        setMotorSpeed(motorC,powerA);
        lasterror = error;
        timer++;
}
```

- We managed to work out the remaining kinks and had Noddy balancing relatively successful with some leeway to occasional errors.
- Conor logged the final notes for the session
- Daniel implemented the notes further into the group learning journal and finalised and compartmentalised notes and plans.