Lab05 Report

Lab05

Section: 4

Bilal Hodzic

10/06/2021

10/06/2021

Problem:

Create a program that determines what side the ds4 is facing and prints it out.

Analysis:

Program tests problem solving ability.

Design: First I created the required functions and tested them by writing temporary code in my main function. I tested until they all worked correctly then begin writing code to get it to output only once. I did this by writing a statement that would store the old value after each print. I then made the entire function only run when the magnitude was below a certain point. This prevented the program from outputting when the magnitude was very high.

Testing:

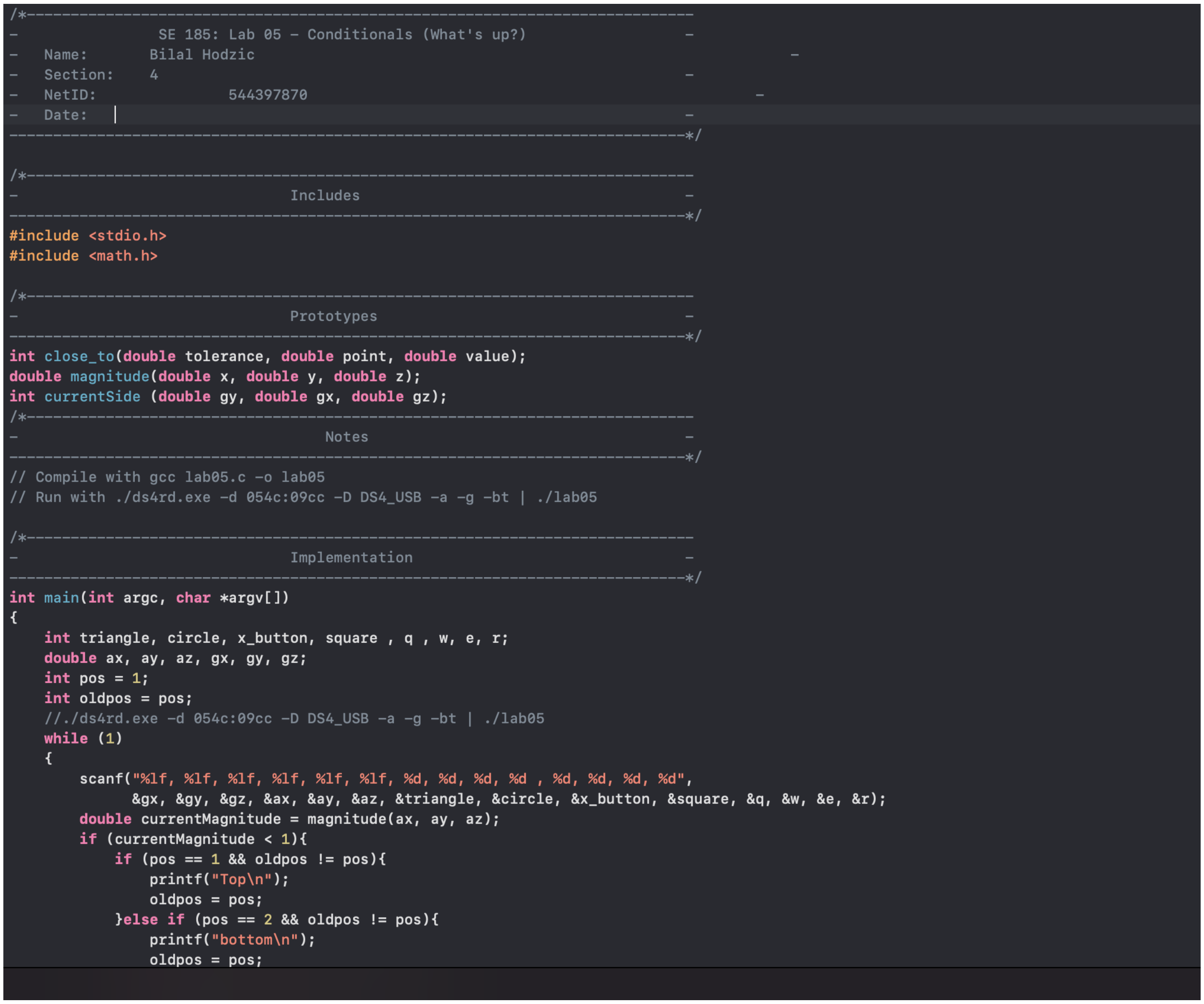
Tested the code frequently to ensure that the components worked.

Comments:

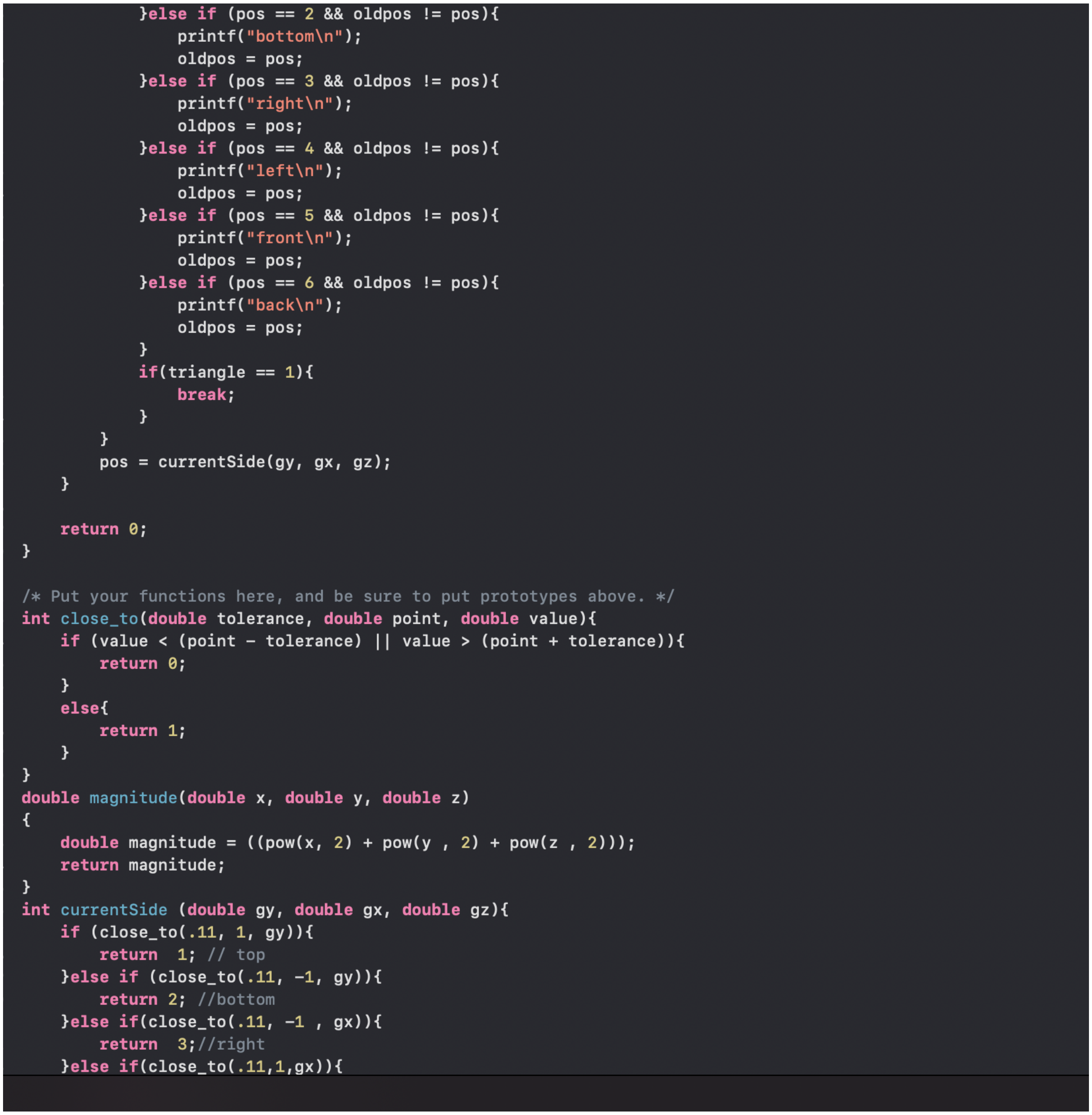
Was a fun challenge.

Screenshots and source code:

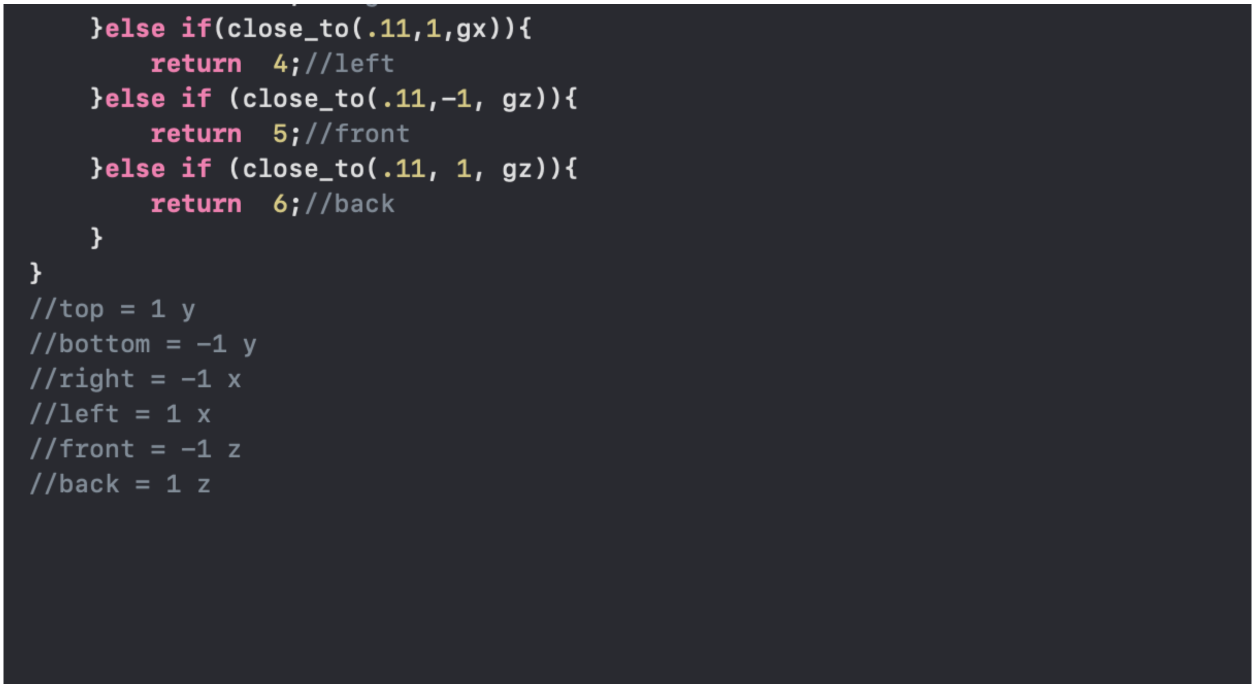
Don't have an output screenshot because I forgot to take one and no longer can at this point.



```
/*-----------------------------------------------------------------------------
-                SE 185: Lab 05 - Conditionals (What's up?)                  -
-   Name:       Bilal Hodzic                                                             -
-   Section:    4                                                            -
-   NetID:               544397870                                                   -
-   Date:                                                                    -
-----------------------------------------------------------------------------*/

/*-----------------------------------------------------------------------------
-                               Includes                                     -
-----------------------------------------------------------------------------*/
#include <stdio.h>
#include <math.h>

/*-----------------------------------------------------------------------------
-                               Prototypes                                   -
-----------------------------------------------------------------------------*/
int close_to(double tolerance, double point, double value);
double magnitude(double x, double y, double z);
int currentSide (double gy, double gx, double gz);
/*-----------------------------------------------------------------------------
-                                   Notes                                    -
-----------------------------------------------------------------------------*/
// Compile with gcc lab05.c -o lab05
// Run with ./ds4rd.exe -d 054c:09cc -D DS4_USB -a -g -bt | ./lab05

/*-----------------------------------------------------------------------------
-                               Implementation                               -
-----------------------------------------------------------------------------*/
int main(int argc, char *argv[])
{
    int triangle, circle, x_button, square , q , w, e, r;
    double ax, ay, az, gx, gy, gz;
    int pos = 1;
    int oldpos = pos;
    //./ds4rd.exe -d 054c:09cc -D DS4_USB -a -g -bt | ./lab05
    while (1)
    {
        scanf("%lf, %lf, %lf, %lf, %lf, %lf, %d, %d, %d, %d , %d, %d, %d, %d",
              &gx, &gy, &gz, &ax, &ay, &az, &triangle, &circle, &x_button, &square, &q, &w, &e, &r);
        double currentMagnitude = magnitude(ax, ay, az);
        if (currentMagnitude < 1){
            if (pos == 1 && oldpos != pos){
                printf("Top\n");
                oldpos = pos;
            }else if (pos == 2 && oldpos != pos){
                printf("bottom\n");
                oldpos = pos;
```



```
            }else if (pos == 2 && oldpos != pos){
                printf("bottom\n");
                oldpos = pos;
            }else if (pos == 3 && oldpos != pos){
                printf("right\n");
                oldpos = pos;
            }else if (pos == 4 && oldpos != pos){
                printf("left\n");
                oldpos = pos;
            }else if (pos == 5 && oldpos != pos){
                printf("front\n");
                oldpos = pos;
            }else if (pos == 6 && oldpos != pos){
                printf("back\n");
                oldpos = pos;
            }
            if(triangle == 1){
                break;
            }
        }
        pos = currentSide(gy, gx, gz);
    }

    return 0;
}

/* Put your functions here, and be sure to put prototypes above. */
int close_to(double tolerance, double point, double value){
    if (value < (point - tolerance) || value > (point + tolerance)){
        return 0;
    }
    else{
        return 1;
    }
}
double magnitude(double x, double y, double z)
{
    double magnitude = ((pow(x, 2) + pow(y , 2) + pow(z , 2)));
    return magnitude;
}
int currentSide (double gy, double gx, double gz){
    if (close_to(.11, 1, gy)){
        return  1; // top
    }else if (close_to(.11, -1, gy)){
        return 2; //bottom
    }else if(close_to(.11, -1 , gx)){
        return  3;//right
    }else if(close_to(.11,1,gx)){
```



```
    }else if(close_to(.11,1,gx)){
        return  4;//left
    }else if (close_to(.11,-1, gz)){
        return  5;//front
    }else if (close_to(.11, 1, gz)){
        return  6;//back
    }
}
//top = 1 y
//bottom = -1 y
//right = -1 x
//left = 1 x
//front = -1 z
//back = 1 z
```