Direction output

LAB 5

SECTION 2

Gavin Monroe

SUBMISSION DATE:

9/27/17

Problem

Problem 1: How did you approach the design?

I felt good about my design. I used 3 functions and kept my code sort and clean. I made sure my code was exactly between two different numbers for each g axis. I made sure to only keep in the code that I used. After I finished my working code I revised it.

Problem 2: What data did you have to read in?

I had to read in the g axis to figure out which way the controller was oriented. The input variables were:

-gx = gyroscope x-axis

-gy = gyroscope y-axis

- gz = gyroscope z-axis

Problem 3: What functions did you choose to implement and why?

```
double mag(double x2, double y2, double z2);
```

Mag was to simply sqrt and my math a little easier when calculating orientation to overall reduce the wide spread numbers.

```
void findDirection(double gxInput, double gyInput, double gzInput, double tole);
```

As the function of the name applies, it's to overall take the inputs of the gyroscope x, y, and z and to change the new orientation ints and to trigger the old with the new.

```
void outputDirection(int oldd, int neww);
```

This function to overall to detect the change of orientation and output the orientation of the controller. This prints TOP, BOTTOM, LEFT, or RIGHT depending on which int it is in.

Problem 4: What tolerance values did you pick and how did you decide on them?

The tolerance had to be very flexible since the accuracy of the gyroscope was pretty good. I chose the numbers by moving the controller in the orientation that It needed it to be in. Then I would move the control a little more just to get the max and low of the x, y, or z. Then I would set my tolerance around that. It worked as I planned it to be.

Analysis

I needed to overall look at the old placement to output the new one.

Design

Nice, simplistic, 3 easy functions.

Testing

Testing I could see that I needed to fflush the code

Comments

This was overall a fun challenge

Source Code

```
/* SE 185 Lab 5 Wrapper Program */

#include <stdio.h>

#include <math.h>

#define TRUE 1

/* Put your lab 4 functions prototypes here, as well as the prototype for lab 5 */

double mag(double x2, double y2, double z2);

void findDirection(double gxInput, double gyInput, double gzInput, double tole);

void outputDirection(int oldd, int neww);

int d;

int d2;

int main(void) {

	int t, b1, b2, b3, b4;

	double ax, ay, az, gx, gy, gz;

	printf("%d %d",d,d2);

	while (TRUE) {

		fflush(stdout);//Flush Code

		scanf("%d, %lf, %lf, %lf, %lf, %lf, %d, %d, %d, %d", &t, &ax, &ay, &az, &gx, &gy, &gz, &b1, &b2,
&b3, &b4 );//Grab Input

	    d2=d;//set old orientation.

		findDirection(gx, gy, gz, 0);//Find Direction which the controller is facing.

		outputDirection(d2, d);//Output Result.

		if (b1!=0){//Close when Triangle is pressed

			return 0;

		}

	}

   return 0;

}

double mag(double x2, double y2, double z2){

	double sqrtX = pow(x2, 2);

	double sqrtY = pow(y2, 2);

	double sqrtZ = pow(z2, 2);

	return sqrt(sqrtX + sqrtY + sqrtZ);

}
```

```
void findDirection(double gxInput, double gyInput, double gzInput, double tole){
	if (gxInput >= 0.014300 && gxInput <= 0.20600){//Top
		if (gyInput >= 0.973900 && gyInput <= 0.995640){
			if (gzInput >= 0.193300 && gzInput <= 0.206687){
				d=1;
			}
		}
	}
	if (gxInput >= 0.014300 && gxInput <= 0.17600){//Bottom
		if (gyInput <= -0.700000 && gyInput >= -0.723900){
			if (gzInput <= -0.720000 && gzInput >= -0.730000){
				d=2;
			}
		}
	}
	if (gxInput >= 0.962000 && gxInput <= 0.99800){//Left
		if (gyInput >= -0.340100 && gyInput <= 0.040500){
			if (gzInput >= 0.221000 && gzInput <= 0.268000){
				d=3;
			}
		}
	}
	if (gxInput <= -0.902000 && gxInput >= -0.99800){//Right
		if (gyInput <= -0.000277 && gyInput >= -0.158900){
			if (gzInput >= 0.21000 && gzInput <= 0.368000){
				d=4;
			}
		}
	}
}
void outputDirection(int oldd, int neww){
	if (oldd!=neww){
		if (neww==1){
			printf("TOP\n");
```

```
                }
                if(neww==2){
                        printf("BOTTOM\n");
                }
                if(neww==3){
                        printf("LEFT\n");
                }
                if(neww==4){
                        printf("RIGHT\n");
                }
        }
}
```

Screen Shots

<I don't have Cygwin on this computer so…./>