Lab07 Report

Lab07

Section: 4

Bilal Hodzic

10/26/2021

10/26/2021

Problem:
Make a program that determines how far the ds4 is being tilted in any given direction and output a bar graph/equalizer effect as a result.
Analysis:
Program reads inputs and outputs based on how it's interpreted.
Design:
First 4 functions were written to read in inputs, scale them to an 80 character screen and output a graph depending on the inputs read. Main method code was then written to use these functions to output a constant graph based on controller tilt. The program also included code that changed whether the program read the roll, pitch, or joystick.
Testing:
Each function was tested individually. The main method was tested after each major addition
Comments:
Bonus 1 is included. The program switches with just one button.
Source code:

```
/*----------------------------------------------------------------------------
-                                                   SE 185 Lab 07 - The DS4 Equalizer
-          Developed for 185-Rursch by T.Tran and K.Wang
-          Name:
-          Section:
-          NetID:
-          Date:
-
-  This file provides the outline for your program
-  Please implement the functions given by the prototypes below and
-  complete the main function to make the program complete.
-  You must implement the functions which are prototyped below exactly
-  as they are requested.
----------------------------------------------------------------------------*/

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


/*----------------------------------------------------------------------------
-                                                                   Defines
----------------------------------------------------------------------------*/
#define PI 3.141592653589

/* NO GLOBAL VARIABLES ALLOWED */


/*----------------------------------------------------------------------------
-                                                                   Prototypes
----------------------------------------------------------------------------*/
/*----------------------------------------------------------------------------
   PRE: Arguments must point to double variables or int variables as appropriate
   This function scans a line of DS4 data, and returns
```

```
   True when left button is pressed
   False Otherwise
   POST: it modifies its arguments to return values read from the input line.
-------------------------------------------------------------------------------*/
int read_input( int* time,
                double* g_x, double* g_y, double* g_z,
                int* button_T, int* button_C, int* button_X, int* button_S,
                int* l_joy_x, int* l_joy_y, int* r_joy_x, int* r_joy_y );

/*-------------------------------------------------------------------------
   PRE: ~(-1.0) <= mag <= ~(1.0)
   This function scales the roll/pitch value to fit on the screen.
   Input should be capped at either -1.0 or 1.0 before the rest of your
   conversion.
   POST: -39 <= return value <= 39
-------------------------------------------------------------------------*/
int scaleMagForScreen(double rad);

/*-------------------------------------------------------------------------
   PRE: -128 <= mag <= 127
   This function scales the joystick value to fit on the screen.
   POST: -39 <= return value <= 39
-------------------------------------------------------------------------*/
int scaleJoyForScreen(int rad);

/*-------------------------------------------------------------------------
   PRE: -39 <= number <= 39
   Uses print_chars to graph a number from -39 to 39 on the screen.
   You may assume that the screen is 80 characters wide.
-------------------------------------------------------------------------*/
void graph_line(int number);

/*-------------------------------------------------------------------------
   PRE: num >= 0
   This function prints the character "use" to the screen "num" times
   This function is the ONLY place printf is allowed to be used
   POST: nothing is returned, but "use" has been printed "num" times
-------------------------------------------------------------------------*/
void print_chars(int num, char use);


/*-------------------------------------------------------------------------
-                                                                Implementation
-------------------------------------------------------------------------*/
int main()
{
   double x, y, z;                  /* Values of x, y, and z axis*/
   int t;                           /* Variable to hold the time value */
   int b_Up, b_Down, b_Left, b_Right;  /* Variables to hold the button statuses */
   int j_LX, j_LY, j_RX, j_RY;         /* Variables to hold the joystick statuses */
   int scaled_pitch, scaled_roll;           /* Value of the roll/pitch adjusted to fit screen display */
   int scaled_joy;                  /* Value of joystick adjusted to fit screen display */

   /* Put pre-loop preparation code here */
```

```
        int old = 1;
        int switchCheck = 0;
        int holdChck = 0;
    do
    {
      /* Scan a line of input */
                if (holdChck == 0){
                        switchCheck += read_input(&t, &x, &y, &z, &b_Up, &b_Right, &b_Down, &b_Left, &j_LX,
&j_LY, &j_RX, &j_RY);
                }
                if(b_Up + b_Down + b_Left > 0){
                        holdChck = 1;
                }else{
                        holdChck = 0;
                }

                if (switchCheck == 0){
                        switchCheck = old;
                }else{
                        old = switchCheck;
                }
                if (switchCheck == 4){
                        switchCheck = 1;
                }
                read_input(&t, &x, &y, &z, &b_Up, &b_Right, &b_Down, &b_Left, &j_LX, &j_LY, &j_RX, &j_RY);

      /* Calculate and scale for pitch AND roll AND joystick */
                scaled_roll = scaleMagForScreen(x);
                scaled_pitch = scaleMagForScreen(z);
                scaled_joy = scaleJoyForScreen(j_RY);

      /* Switch between roll, pitch, and joystick with the up, down, and right button, respectivly */
                if (switchCheck == 1){
                        graph_line(scaled_roll);
                }else if (switchCheck == 2){
                        graph_line(scaled_pitch);
                }else if (switchCheck == 3){
                        graph_line(scaled_joy);
                }else if (switchCheck >= 5){
                        break;
                }
      /* Output your graph line */


      fflush(stdout);

    } while (1 ); /* Modify to stop when left button is pressed */

    return 0;

}

int read_input( int* time,
            double* g_x, double* g_y, double* g_z,
```

```
                int* button_T, int* button_C, int* button_X, int* button_S,
                int* l_joy_x, int* l_joy_y, int* r_joy_x, int* r_joy_y)
{
        int q, w, e, r;
        scanf(" %d, %lf, %lf, %lf, %d, %d, %d, %d, %d, %d, %d, %d, %d, %d, %d, %d", time, g_x, g_y, g_z,
button_T, button_C,
                                button_X, button_S, &q, &w, &e, &r, l_joy_x, l_joy_y, r_joy_x, r_joy_y);
        if (*button_T == 1 || *button_C == 1 || *button_X == 1){
                return 1;
        }else if (*button_S == 1){
                return 5;
        }else {
                return 0;
        }
}

int scaleMagForScreen(double rad){
        return rad * 39;
}
int scaleJoyForScreen(int rad){
        return (rad /3.25) * -1;
}
void graph_line(int number){
        if (number > 0){
                char use  = 'L';
                print_chars(number, use);

        }else{
                number *= -1;
                char use = 'R';
                print_chars(number, use);
        }
}
void print_chars(int num, char use){
        if (num == 0){
                printf("                                0");
                printf("\n");
        }else if (use == 'R'){
                printf("                                 ");
                for (int i = 0; i < num; i++){
                        printf("%c", use);
                }
                printf("\n");
        }else{
                for (int i = 0; i < 39-num; i++){
                        printf(" ");
                }
                for (int i = 0; i < num; i++){
                        printf("L");
                }
                printf(" ");
                printf("\n");
        }
}
```

Output Screenshots: