TITLE

Introduction the DS4 and Functions

LAB # 03

SECTION # 8

FULL NAME

Adam Jennissen

SUBMISSION DATE:

9/18/2022

DATE

9/15/2022

Problem

The purpose of this lab was to create 2 programs that interacted with a DualShock 4 controller as well as learn how to use the DualShock 4 controllers inside of Cygwin. The first program converted milliseconds into minutes, seconds, and milliseconds. The second program counted the number of buttons pressed on the DualShock 4 controller.

Analysis

The program had to take input from the ds4 controller and output it to a spreadsheet, as well as convert the time to minutes, seconds, and milliseconds. For this I had to know that there are 60000 milliseconds in a minute, and 1000 milliseconds in a minute. For the second problem I just had to take input from the ds4 controller and perform basic addition.

Design

For the first program I had to figure out how to convert from milliseconds into minutes, seconds, and milliseconds. I first converted milliseconds to minutes by dividing by 60000 and using integer division to my advantage. In order to find the seconds, I used the remainder of 60000 to find the milliseconds left after converting to minutes, and then divided it by 1000 to get the seconds using integer division. Lastly to get the milliseconds left, I used the remainder of 60000 and the remainder of 1000, to get the millisecond left after finding both minutes and seconds. For the second problem, I had to take the input, which I just looked at the other problem to figure out how to do. I then had to create a program that used that input and then added the number of buttons pressed, which was simple and just involved adding the numbers together.

Testing

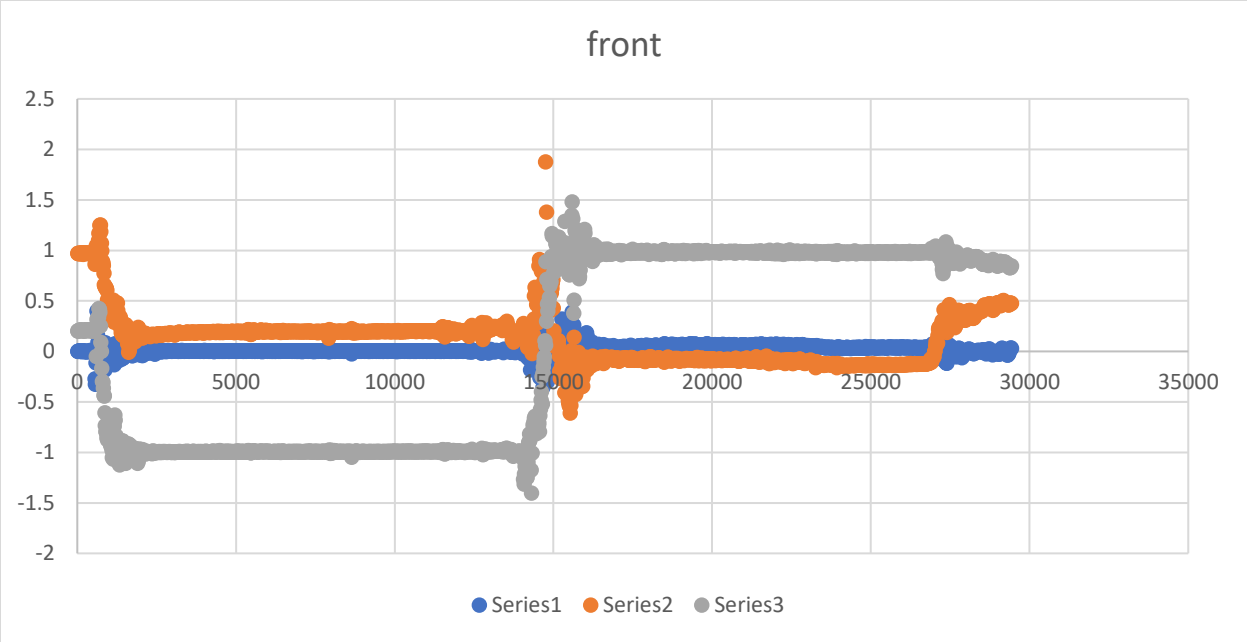
In order to make sure that my programs outputted the correct thing, I had to run them for a while to make sure they worked right. For the first problem, I ran it for about a minute to make sure that all of my math worked correctly. In the screenshot it only ran for about 30 seconds (4). When testing the second problem I tried a combination of all the different buttons, to make sure that it would add any of them together (5).

Comments

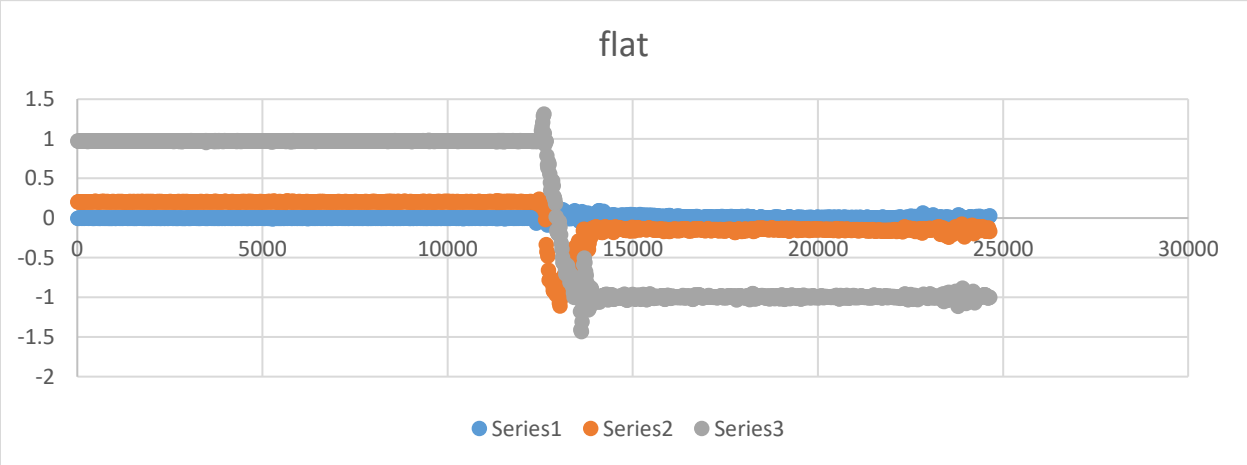
I learned how to find the device ID of a ds4 controller, and then how to use a ds4 controller through Cygwin. I also learned multiple different tags that can be used, depending on what you want to track form the ds4 controller. I got a better understanding of the modulo operator when converting milliseconds to minutes, seconds, and milliseconds. Doing both of the problems also gave me a better understanding of how functions work, and how the program actually runs.

Screen Shots

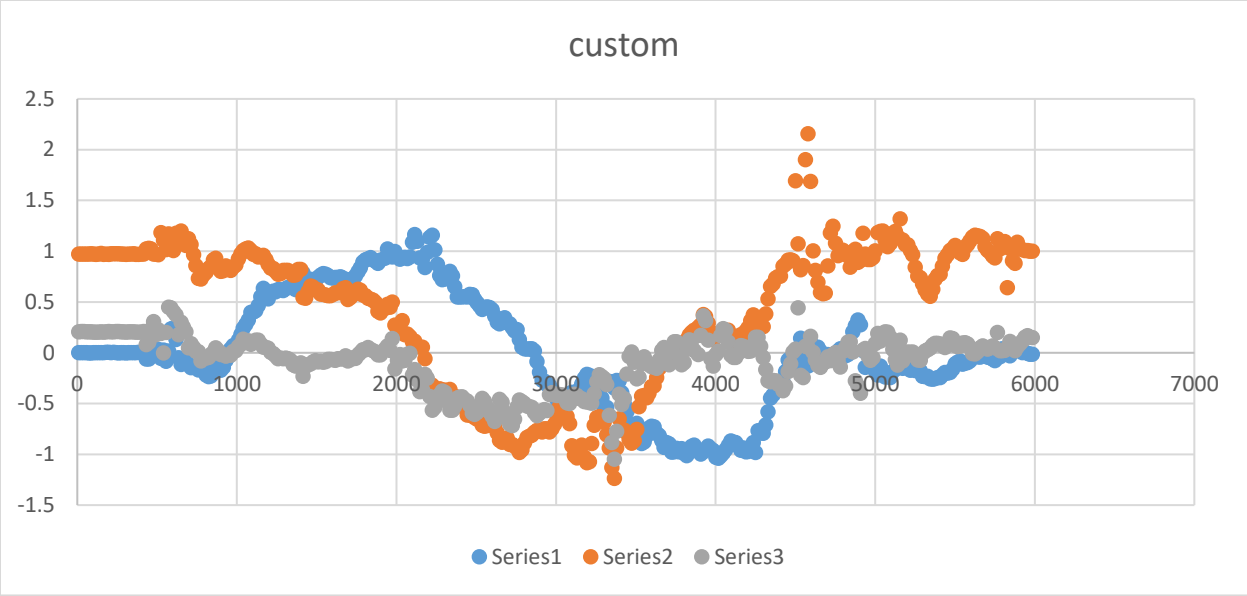
1.


front
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
5000
10000
15000
20000
25000
30000
35000
Series1
Series2
Series3

2.


flat
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
5000
10000
15000
20000
25000
30000
Series1
Series2
Series3

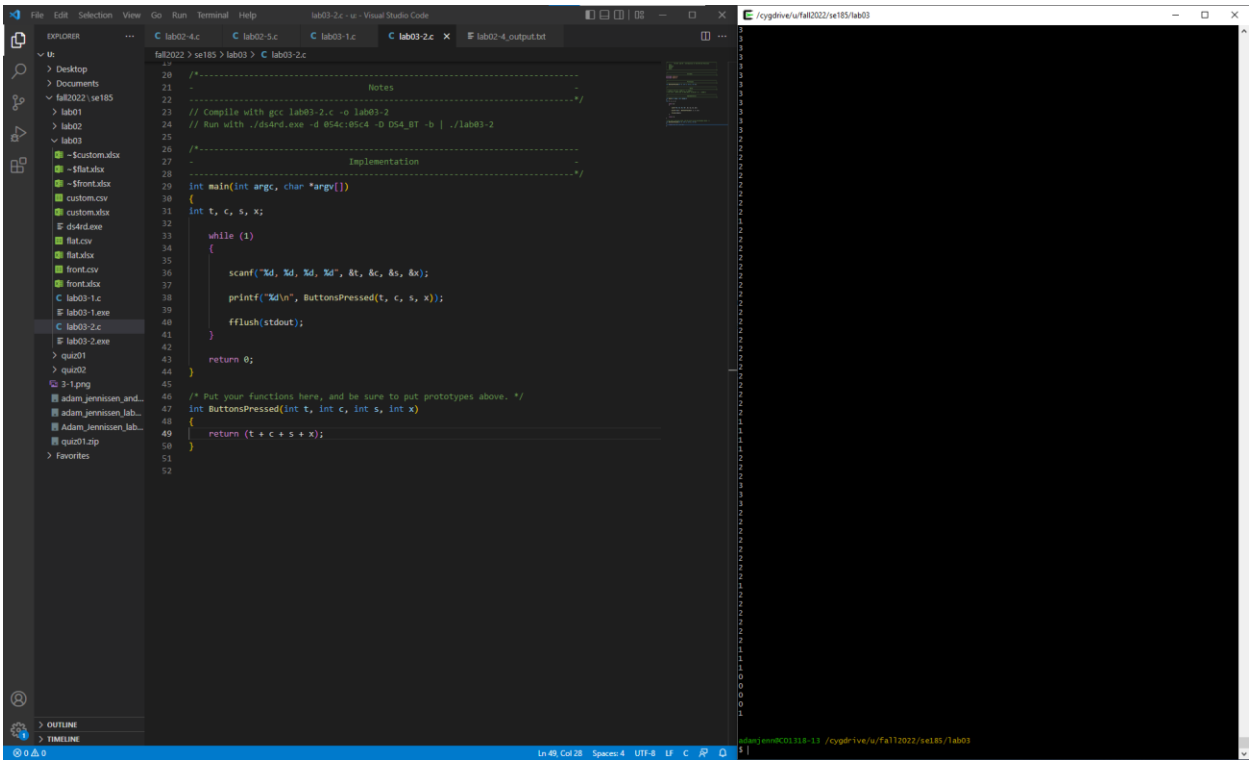
3.



4.

```
adam jennissen and...   72      return ((t % 60000) % 1000);
/cygdrive/u/fall2022/se185/lab03
At 29476 ms, the acceleration's magnitude was: 0.004711
At 0 minutes, 29 seconds, and 476 milliseconds it was: 0.004711
Echoing output:   29.491, -0.0020, -0.0010, -0.0044
At 29491 ms, the acceleration's magnitude was: 0.004910
At 0 minutes, 29 seconds, and 491 milliseconds it was: 0.004910
Echoing output:   29.507, -0.0015, -0.0010, -0.0039
At 29507 ms, the acceleration's magnitude was: 0.004287
At 0 minutes, 29 seconds, and 507 milliseconds it was: 0.004287
Echoing output:   29.523, -0.0015, -0.0015, -0.0039
At 29523 ms, the acceleration's magnitude was: 0.004424
At 0 minutes, 29 seconds, and 523 milliseconds it was: 0.004424
Echoing output:   29.538, -0.0024, -0.0015, -0.0049
At 29538 ms, the acceleration's magnitude was: 0.005655
At 0 minutes, 29 seconds, and 538 milliseconds it was: 0.005655
Echoing output:   29.554, -0.0015, -0.0015, -0.0049
At 29554 ms, the acceleration's magnitude was: 0.005307
At 0 minutes, 29 seconds, and 554 milliseconds it was: 0.005307
Echoing output:   29.570, -0.0015, -0.0010, -0.0039
At 29570 ms, the acceleration's magnitude was: 0.004287
At 0 minutes, 29 seconds, and 570 milliseconds it was: 0.004287
Echoing output:   29.585, -0.0020, -0.0015, -0.0034
At 29585 ms, the acceleration's magnitude was: 0.004203
At 0 minutes, 29 seconds, and 585 milliseconds it was: 0.004203
Echoing output:   29.601, -0.0015, -0.0024, -0.0024
At 29601 ms, the acceleration's magnitude was: 0.003753
At 0 minutes, 29 seconds, and 601 milliseconds it was: 0.003753
Echoing output:   29.616, -0.0020, -0.0005, -0.0034
At 29616 ms, the acceleration's magnitude was: 0.003969
At 0 minutes, 29 seconds, and 616 milliseconds it was: 0.003969
Echoing output:   29.632, -0.0020, -0.0015, -0.0034
At 29632 ms, the acceleration's magnitude was: 0.004203
At 0 minutes, 29 seconds, and 632 milliseconds it was: 0.004203
Echoing output:   29.648, -0.0020, -0.0015, -0.0039
At 29648 ms, the acceleration's magnitude was: 0.004609
At 0 minutes, 29 seconds, and 648 milliseconds it was: 0.004609
Echoing output:   29.663, -0.0020, -0.0010, -0.0034
At 29663 ms, the acceleration's magnitude was: 0.004058
At 0 minutes, 29 seconds, and 663 milliseconds it was: 0.004058
Echoing output:   29.679, -0.0015, -0.0010, -0.0039
At 29679 ms, the acceleration's magnitude was: 0.004287
At 0 minutes, 29 seconds, and 679 milliseconds it was: 0.004287
Echoing output:   29.694, -0.0020, -0.0015, -0.0039
At 29694 ms, the acceleration's magnitude was: 0.004609
At 0 minutes, 29 seconds, and 694 milliseconds it was: 0.004609
Echoing output:   29.710, -0.0010, -0.0010, -0.0044
At 29710 ms, the acceleration's magnitude was: 0.004609
At 0 minutes, 29 seconds, and 710 milliseconds it was: 0.004609
```

5.


File Edit Selection View Go Run Terminal Help
lab03-2.c - u: - Visual Studio Code
EXPLORER
lab02-4.c
lab02-5.c
lab03-1.c
lab03-2.c
lab02-4_output.txt
fall2022 > se185 > lab03 > lab03-2.c
/*----------
- Notes -
----------*/
// Compile with gcc lab03-2.c -o lab03-2
// Run with ./ds4rd.exe -d 054c:05c4 -D DS4_BT -b | ./lab03-2
/*----------
- Implementation -
----------*/
int main(int argc, char *argv[])
{
int t, c, s, x;
while (1)
{
scanf("%d, %d, %d, %d", &t, &c, &s, &x);
printf("%d\n", ButtonsPressed(t, c, s, x));
fflush(stdout);
}
return 0;
}
/* Put your functions here, and be sure to put prototypes above. */
int ButtonsPressed(int t, int c, int s, int x)
{
return (t + c + s + x);
}
Ln 49, Col 28 Spaces: 4 UTF-8 LF C
/cygdrive/u/fall2022/se185/lab03