

# **Introductions to Functions**

## **LAB 3**

### **SECTION G**

**SUBMITTED BY:**

**HARI BISWA**

**SUBMISSION DATE:**

**9/7/2018**

## **Problem**

The purpose of this lab is to understand the concept of functions and create a function, practicing the writing mod and integer division expressions and formatting output practice. In this activity specifically, I was assigned to create a function to convert the milliseconds to seconds to minutes precisely. I also was assigned to write a code that's going to output the numbers like 1,2,3 and 4 if we pressed the shapes buttons on the Blackboard/controller.

## **Analysis**

There is a specific constraint that are given to code. On this lab which is lab 3, I was given the formula which is a Pythagorean theorem formula which is also named as the mag function. In this case the mag function which is the Pythagorean theorem is the input which is in the milliseconds and it returned it to "the acceleration's magnitude was 1.00037356 or 0.8479747, it is also shown in the picture below in the "Capture 2." After finding the Mag function, I had to input the specific milliseconds which was the "129313" and output it into the more readable form which is 2 min, 9 sec and 313 milliseconds.

## **Design**

The goal was to output the 2 min, 9 sec, 313 milliseconds. To approach this, I created a three different functions for minutes, seconds and milliseconds. To convert from milli to seconds, I divide the milli by 1000. To get the minutes, I divide the milli (129313) by  $1000 \times 60$  and  $1000 \times 60$  must be inside the parenthesis in order to get to 2 min. For the seconds, instead creating more variables and functions, I just put the milli (129313) module (%) 1000 and it gave me 313 because the module only takes the remainder in C.

## **Testing**

I was failed to do this task most of the time because it keeps saying errors on every line of the code but then looking very carefully to fix the errors was very time extending but it was worth it because I was able to fix. I also ran into the computer help hours for help because I was very stuck on the part four but then I figured it was very easy after they helped with it because it was just the semicolon and comma errors. After I went to the help hours, I was able to finish and complete the task successful.

## **Comments**

In doing this part of the lab, I learned to ask for help without hesitation because when I went to the help hours for they help me with a creating functions faster as well.

**/\* 185 Lab 3 Template \*/**

**PART 1-3**

**/\* 185 Lab 3 Template \*/**

#include <stdio.h>

#include <math.h>

**/\* Put your function prototypes here \*/**

double mag(double ax, double ay, double az);

int minutes(int t);

int seconds(int t);

int millis(int t);

int main(void) {

**/\* DO NOT MODIFY THESE VARIABLE DECLARATIONS \*/**

int t;

double ax, ay, az;

**/\* This while loop makes your code repeat. Don't get rid of it. \*/**

while (1) {

scanf("%d,%lf,%lf,%lf", &t, &ax, &ay, &az);

**/\* CODE SECTION 0 \*/**

//int s = t/1000;

// printf("Echoing output: %8.3d, %7.4lf, %7.4lf, %7.4lf\n", s, ax, ay, az);

**/\* CODE SECTION 1 \*/**

//printf("At %d ms, the acceleration's magnitude was: %lf\n",

// t, mag(ax, ay, az));

**/\* CODE SECTION 2 \*/**

printf("At %d minutes, %d seconds, and %d milliseconds it was: %lf\n",

minutes(t), seconds(t), millis(t), mag(ax,ay,az));

double c = mag(ax, ay, az) ;

}

return 0;

}

**/\* Put your functions here \*/**

```
double mag(double ax,double ay, double az){
double magnet = sqrt(pow(ax,2)+pow(ay,2)+pow(az,2));
```

```
    return magnet;
```

```
}
```

```
int minutes(int t){
    int min = t/(1000*60);
    return min;
```

```
}
```

```
int seconds(int t){
    int sec = t/1000;
    sec = sec % 60;
```

```
    return sec;
```

```
}
```

```
int millis(int t){
    int millisec = t%1000;
    return millisec;
```

```
}
```

#### **PART 4**

```
#include <stdio.h>
```

```
#include <math.h>
```

```
int Count(int a,int b,int c,int d);
```

```
int main (void) {
```

```
    int t,c,x,s;
```

```
/* This while loop makes your code repeat. Don't get rid of it. */
```

```
    while (1) {
```

```
        scanf("%d,%d,%d,%d", &t ,&c ,&x ,&s);
```

```
        printf("This is the amount of buttons pressed: %d\n", Count(t,c,x,s));
```

```
        fflush(stdout);
```

```
    }
```

```
return 0;
```

```
}
```

```
int Count(int a,int b,int c,int d){
```

```
    return a+b+c+d;
```

```
}
```

#### **Capture 1**

```

echoing output: 015, 0.0209, 0.9814, 0.2144
echoing output: 015, 0.0264, 0.9770, 0.2072
echoing output: 015, 0.0316, 0.9717, 0.2125
echoing output: 015, 0.0291, 0.9749, 0.2142
echoing output: 015, 0.0280, 0.9807, 0.2116
echoing output: 015, 0.0259, 0.9807, 0.2050
echoing output: 015, 0.0268, 0.9791, 0.2061
echoing output: 015, 0.0252, 0.9801, 0.2076
echoing output: 015, 0.0273, 0.9806, 0.2045
echoing output: 015, 0.0240, 0.9794, 0.2093
echoing output: 015, 0.0286, 0.9772, 0.2115
echoing output: 015, 0.0297, 0.9756, 0.2169
echoing output: 015, 0.0240, 0.9731, 0.2075
echoing output: 015, 0.0242, 0.9772, 0.2133
echoing output: 015, 0.0248, 0.9808, 0.2153
echoing output: 015, 0.0230, 0.9784, 0.2124
echoing output: 015, 0.0251, 0.9770, 0.2104
echoing output: 015, 0.0275, 0.9847, 0.2080
echoing output: 015, 0.0253, 0.9774, 0.2116
echoing output: 015, 0.0264, 0.9788, 0.2148
echoing output: 015, 0.0220, 0.9791, 0.2125
echoing output: 015, 0.0255, 0.9751, 0.2071
echoing output: 015, 0.0306, 0.9792, 0.2120
echoing output: 015, 0.0274, 0.9763, 0.2106
echoing output: 015, 0.0224, 0.9750, 0.2047
echoing output: 015, 0.0247, 0.9825, 0.1998
echoing output: 015, 0.0261, 0.9742, 0.2142
echoing output: 015, 0.0278, 0.9824, 0.2042
echoing output: 015, 0.0244, 0.9802, 0.2095
echoing output: 015, 0.0237, 0.9753, 0.2125
echoing output: 015, 0.0242, 0.9749, 0.2109
echoing output: 015, 0.0275, 0.9752, 0.2146
echoing output: 015, 0.0287, 0.9755, 0.2135
echoing output: 015, 0.0294, 0.9769, 0.2065
echoing output: 015, 0.0306, 0.9774, 0.2094
echoing output: 015, 0.0242, 0.9792, 0.2084
echoing output: 015, 0.0252, 0.9806, 0.2031
echoing output: 015, 0.0253, 0.9792, 0.2120
echoing output: 015, 0.0224, 0.9681, 0.2185
echoing output: 015, 0.0280, 0.9780, 0.2130
echoing output: 015, 0.0261, 0.9795, 0.2037
echoing output: 015, 0.0269, 0.9762, 0.2043
echoing output: 015, 0.0228, 0.9755, 0.2031
echoing output: 015, 0.0224, 0.9747, 0.2055
echoing output: 015, 0.0272, 0.9791, 0.2128
echoing output: 015, 0.0255, 0.9761, 0.2143
echoing output: 015, 0.0292, 0.9827, 0.2122
echoing output: 016, 0.0223, 0.9714, 0.2104
echoing output: 016, 0.0270, 0.9772, 0.2113
echoing output: 016, 0.0266, 0.9769, 0.2125
echoing output: 016, 0.0246, 0.9758, 0.2163
echoing output: 016, 0.0251, 0.9788, 0.2109
echoing output: 016, 0.0250, 0.9769, 0.2071
echoing output: 016, 0.0248, 0.9779, 0.2111
echoing output: 016, 0.0251, 0.9742, 0.2077
echoing output: 016, 0.0300, 0.9789, 0.2102
echoing output: 016, 0.0280, 0.9735, 0.2105
echoing output: 016, 0.0230, 0.9750, 0.2115
echoing output: 016, 0.0236, 0.9730, 0.2126
echoing output: 016, 0.0296, 0.9800, 0.2082
echoing output: 016, 0.0287, 0.9778, 0.2069
echoing output: 016, 0.0279, 0.9799, 0.2103
echoing output: 016, 0.0268, 0.9774, 0.2135
echoing output: 016, 0.0239, 0.9819, 0.2086
echoing output: 016, 0.0275, 0.9760, 0.2125
echoing output: 016, 0.0256, 0.9724, 0.2122

```

```

biswa17@C02018-22 /cygdrive/u/CprE185/Lab3
$ |

```

## Capture 2

```

At 10834 ms, the acceleration's magnitude was: 0.999254
At 10844 ms, the acceleration's magnitude was: 1.001874
At 10854 ms, the acceleration's magnitude was: 1.000990
At 10874 ms, the acceleration's magnitude was: 0.997601
At 10894 ms, the acceleration's magnitude was: 1.005586
At 10904 ms, the acceleration's magnitude was: 1.002071
At 10914 ms, the acceleration's magnitude was: 1.000724
At 10934 ms, the acceleration's magnitude was: 1.005376
At 10944 ms, the acceleration's magnitude was: 0.996675
At 10965 ms, the acceleration's magnitude was: 1.000988
At 10984 ms, the acceleration's magnitude was: 0.996076
At 10994 ms, the acceleration's magnitude was: 1.009797
At 11004 ms, the acceleration's magnitude was: 0.997043
At 11014 ms, the acceleration's magnitude was: 1.007121
At 11024 ms, the acceleration's magnitude was: 1.002076
At 11044 ms, the acceleration's magnitude was: 0.993802
At 11064 ms, the acceleration's magnitude was: 1.000013
At 11084 ms, the acceleration's magnitude was: 1.000897
At 11094 ms, the acceleration's magnitude was: 1.001530
At 11104 ms, the acceleration's magnitude was: 1.000381
At 11114 ms, the acceleration's magnitude was: 0.996736
At 11134 ms, the acceleration's magnitude was: 0.998025
At 11144 ms, the acceleration's magnitude was: 0.997948
At 11164 ms, the acceleration's magnitude was: 0.998233
At 11184 ms, the acceleration's magnitude was: 1.004207
At 11194 ms, the acceleration's magnitude was: 0.996944
At 11214 ms, the acceleration's magnitude was: 1.002303
At 11234 ms, the acceleration's magnitude was: 1.003422
At 11244 ms, the acceleration's magnitude was: 1.001692
At 11264 ms, the acceleration's magnitude was: 0.996690
At 11274 ms, the acceleration's magnitude was: 0.999065
At 11294 ms, the acceleration's magnitude was: 0.998610
At 11304 ms, the acceleration's magnitude was: 0.999386
At 11314 ms, the acceleration's magnitude was: 0.999005
At 11334 ms, the acceleration's magnitude was: 0.997852
At 11344 ms, the acceleration's magnitude was: 0.999800
At 11364 ms, the acceleration's magnitude was: 1.002392
At 11384 ms, the acceleration's magnitude was: 0.998726
At 11394 ms, the acceleration's magnitude was: 1.001560
At 11414 ms, the acceleration's magnitude was: 1.000461
At 11424 ms, the acceleration's magnitude was: 0.998563
At 11444 ms, the acceleration's magnitude was: 1.001178
At 11454 ms, the acceleration's magnitude was: 0.996842
At 11464 ms, the acceleration's magnitude was: 1.000925
At 11484 ms, the acceleration's magnitude was: 0.999374
At 11494 ms, the acceleration's magnitude was: 1.001990
At 11504 ms, the acceleration's magnitude was: 0.996643
At 11514 ms, the acceleration's magnitude was: 1.001342
At 11534 ms, the acceleration's magnitude was: 0.995503
At 11544 ms, the acceleration's magnitude was: 0.999926
At 11564 ms, the acceleration's magnitude was: 0.999874
At 11584 ms, the acceleration's magnitude was: 1.000241
At 11594 ms, the acceleration's magnitude was: 1.002520
At 11604 ms, the acceleration's magnitude was: 0.999452
At 11614 ms, the acceleration's magnitude was: 0.998202
At 11634 ms, the acceleration's magnitude was: 0.997997
At 11644 ms, the acceleration's magnitude was: 0.998013
At 11664 ms, the acceleration's magnitude was: 1.005555
At 11684 ms, the acceleration's magnitude was: 0.995798
At 11694 ms, the acceleration's magnitude was: 0.995130
At 11704 ms, the acceleration's magnitude was: 1.002372
At 11724 ms, the acceleration's magnitude was: 0.998226
At 11744 ms, the acceleration's magnitude was: 1.000926
At 11764 ms, the acceleration's magnitude was: 1.001072
At 11784 ms, the acceleration's magnitude was: 0.991998
At 11794 ms, the acceleration's magnitude was: 1.000909
At 11814 ms, the acceleration's magnitude was: 0.997632

```

```

hbiswa17@C02018-22 /cygdrive/u/CprE185/Lab3
$ sni|

```

## Capture 3

```

hbiswa17@vdi-1009-22 /cygdrive/u/CprE185/Lab3
$ gcc lab3.c -o lab3

```

```

hbiswa17@vdi-1009-22 /cygdrive/u/CprE185/Lab3
$ ./lab3

```

```
129313
```

```
At 2 minutes, 9 seconds, and 313 milliseconds it was: 0.000000
```

```
|
```

## Capture 4

This is the amount of buttons pressed: 1This is the amount of buttons pressed: 1This is the amount of buttons pressed: 1

## Capture 5

2This is the amount of buttons pressed: 2This is the amount of buttons pressed: 2This is the amount of buttons pressed: 2`  
2This is the amount of buttons pressed: 2This is the amount of buttons pressed: 2This is the amount of buttons pressed: 2`

### Capture 6

3This is the amount of buttons pressed: 3This is the amount of buttons pressed: 3This is the amount of buttons pressed: 3`  
3This is the amount of buttons pressed: 3This is the amount of buttons pressed: 3This is the amount of buttons pressed: 3`

### Capture 7

4This is the amount of buttons pressed: 4This is the amount of buttons pressed: 4This is the amount of buttons pressed: 4`  
4This is the amount of buttons pressed: 4This is the amount of buttons pressed: 4This is the amount of buttons pressed: 4`

# Or

This is the amount of buttons pressed: 1This is the amount of buttons pressed: 1`  
This is the amount of buttons pressed: 1This is the amount of buttons pressed: 1`  
This is the amount of buttons pressed: 2This is the amount of buttons pressed: 2`  
This is the amount of buttons pressed: 2This is the amount of buttons pressed: 2`  
This is the amount of buttons pressed: 3This is the amount of buttons pressed: 3`  
This is the amount of buttons pressed: 3This is the amount of buttons pressed: 3`  
This is the amount of buttons pressed: 4This is the amount of buttons pressed: 4`  
This is the amount of buttons pressed: 4This is the amount of buttons pressed: 4`  
This is the amount of buttons pressed: 1This is the amount of buttons pressed: 1`  
This is the amount of buttons pressed: 0This is the amount of buttons pressed: 0`  
This is the amount of buttons pressed: 0This is the amount of buttons pressed: 0`  
This is the amount of buttons pressed: 0This is the amount of buttons pressed: 0`