



Introduction to Computing CS 151

Department of Physics and Computer Science
Medgar Evers College

Exam 3

Direction: Submit your typed work(s) as an upload(s) to the Exams directory of your GitHub repository or Dropbox, or in your Exam03 google classroom assignment.

Section	Maximum Points	Points Earned
Fundamentals	5	
Problem Solving	5	
Tracing	5	
Debugging	5	
Total	20	

Fundamentals

- For each of the following questions, write **ONLY** what is requested.
 - Define an int function named `M()` that takes three int parameters. It should return the value of the median of the parameters (second largest value).
 - Write a statement that repeats the message "I Can Do This." on separate lines indefinitely.
 - Define a bool function named `T()` that takes an int reference parameter. If the parameter is a multiple of 6, the function assigns the parameter triple its value; and then, return true; otherwise, the function assigns the parameter half its value and returns false.
 - Write a statement that displays the multiples of 7 between 1 and 100 on their own lines in reverse order.
 - Define a double function named `A()` that takes a double parameter. It should prompt the user to enter a number; and then, return the average of the input of the user and the parameter.

Problem Solving

- Write a void function named `TimesTable()` that takes an int parameter. If the parameter is a positive number less than or equal to 833, the function displays a times table that has the parameter number of rows and 12 columns including labels for the columns. Each row will display the first 12 multiples of a number starting from 1 in sequential order. All numbers will display as four digit numbers with 0 padded to the beginning if necessary. There will be a space between each number in a row and the columns must align. For instance, the first two lines given that the parameter is greater than 1, would display as

```
0001 0001 0002 0003 0004 0005 0006 0007 0008 0009 0010 0011 0012
0002 0002 0004 0006 0008 0010 0012 0014 0016 0018 0020 0022 0024
```

Define any additional function you find necessary.

Tracing

3. Generate the first 18 steps for the trace table or trace table list of the function call SS(4) with the below definition and determine what will be returned when the function call terminates.

```
int SS(int n)
{
    if(n < 0)
    {
        n *= -1;
    }
    int t = 1, s;

    for(int i = 2; i <= n; i += 1)
    {
        s = 1 + i;

        for(int j = 2; j < i; j += 1)
        {
            if(i % j == 0)
            {
                s += j;
            }
        }
        t += s;
    }
    return t;
}
```

Debugging

4. Write **ONLY** the line number and the entire line correction for each line that has an error in the code below.

```
01 int CF(int& n,int b)
02 {
03     c = 0;
04
05     while(n % b == 0)
06     {
07         n / b = n;
08         c = c + 1;
09     }
10     return C;
11 }
12
13 bool IP(int p);
14 {
15     for(int i = 2;i ^ 2 < p;i += 1)
16     {
17         if(p % i == 0)
18         {
19             return false;
20         }
21     }
22     return True;
23 }
24
25 void PF(n)
26 {
27     int m = (n < 0)?(n * -1):(n);
28     int p = 2;
29
30     while(m > 1)
31     {
32         if(IP(p) & m % p == 0)
33         {
34             string c = CF(m,p);
35             cout << p << " " << c;
36
37             if(m > 1)
38             {
39                 cout << " ";
40             }
41         }
42         p =+ 1;
43     }
44 }
```