

CS 31 LA Worksheet Week 3

This worksheet is entirely **optional**, and meant to prepare you for upcoming projects and exams. Some problems will be more challenging than others and are designed to have you apply your knowledge beyond the examples presented in lecture, discussion or projects. We encourage you to collaborate with your peers while completing this worksheet. If you finish the worksheet early, you are welcome to inquire with your LA for this week's Supplemental Problems.

If you have any questions or concerns please contact your LA or go to any of the LA office hours.

Concepts: Loops, If Statements, Cin, Variables, Doubles, Ints

Reading Problems

1. This code snippet tries to print all prime numbers between 3 (inclusive) and a given input n (exclusive). Find the **3 bugs** contained in the code and fix them.

```
int n;
cin >> n;
for (int candidate = 3; candidate < n; ++candidate) {
    bool isPrime = true;
    for (int x = 2; x < n; x++) {
        if (candidate % x == 0){
            isPrime = false;
        }
    }

    if (isPrime) {
        cout << n << " ";
    }
}
```

Programming Problems

1. Write a program that takes in a number as an int and outputs the sum of all of the digits in that number.

Sample Output:

Enter a number: 184

The sum of the digits in your number is 13!

2. Write a program that takes in N numbers and writes their mean as a double.

Sample output:

How many numbers do you want to average? 5

Number: 4

Number: 2

Number: 8

Number: 9

Number: 7

The average is 6!

3. Write a program that reads in an integer N and prints an NxN box where the (i,j)th character is as follows:

' ' if $j > i$
i + j otherwise

Where i is the row number and j is the column number (starting at 0, not 1). For Example, if the input is 4, it should print:

```
0 . . .
1 2 . .
2 3 4 .
3 4 5 6
```

Bonus:

Without using any explicit if/else statements, can you modify the previous program to only print out the sums and no ' 's?

Example, if the input is 4, it should print:

```
0
1 2
2 3 4
3 4 5 6
```

4. Write a program that reads in an integer and prints whether that number is a perfect number. A perfect number is defined as a number that is equal to the sum of all positive factors excluding itself.

Example:

$4 \neq 1 + 2$	=> Print "Not perfect."
$5 \neq 1$	=> Print "Not perfect."
$6 = 1 + 2 + 3$	=> Print "Perfect."
$12 \neq 1 + 2 + 3 + 4 + 6$	=> Print "Not perfect."
$28 = 1 + 2 + 4 + 7 + 14$	=> Print "Perfect."

5. Write a program that takes in an integer N where $N > 0$, and outputs all its factors, each one separated by a comma. There should be no comma before the first number or after the last one.

Sample input:

12

Sample output:

1,2,3,4,6,12

6. Write a program that, given an input integer N, finds an integer x such that $2^x \leq N < 2^{x+1}$. The program should ask for user input and print the integer x it finds. If there is no such x, it should print "error" and nothing more.

Sample Input:

200 => Should output 7, since $2^7 = 128 \leq 200 < 2^8 = 256$.

20 => Should output 4, since $2^4 = 16 \leq 20 < 2^5 = 32$.

8 => Should output 3, since $2^3 = 8 \leq 8 < 2^4 = 16$.

7. The Fibonacci series consists of the integers 0, 1, 1, 2, 3, 5, 8, With the initial values $n_1 = 0$ and $n_2 = 1$ it is possible to find the next number, because the next number is related to the preceding two by the formula $x_{n+1} = x_n + x_{n-1}$. For example, $1+1 = 2$, the next number in the series. Based on this information, write a program that receives an integer n as an input and prints the n^{th} Fibonacci number. **What is the 10th one?**

Your program should also check whether the integer provided is valid. If the user inputs zero or a negative number, the program should print *Error: The input must be positive* and nothing more.

If you haven't done so already, try to write the program using a do-while loop.