# PDS Lab - Section 9

Assignment 3

January 27th, 2020

**Fine Naming:** If you roll no. is 18CS30012, the file name would be 18CS30012_p1.c, 18CS30012_p2.c and 18CS30012_p3.c.

1. Write a program to randomly generate a sequence of integers between 5 and 95 (both inclusive) and output the maximum and minimum values generated so far. If an integer less than 10 is generated, the program should exit. While exiting, report the mean of the numbers generated so far (including the integer less than 10). **You must not store the sequence generated, but update the maximum, minimum and mean, as soon as a new entry is generated.**

   You should use the library function *rand*() for generating a random number. Note that rand() generates a number between 0 and a maximum value $MAX$. To generate a random number between $A$ and $B$, you can use

   ```
   #include <stdlib.h> //in the header
   int randInt = (rand() % (B+1-A)) + A; //to generate a random integer
   ```

   **Example Output:**

   ```
   Iteration 1: Number 43: maximum = 43, minimum = 43, mean = 43
   Iteration 2: Number 21: maximum = 43, minimum = 21, mean = 32
   Iteration 3: Number 5: maximum = 43, minimum =5, mean = 23
   ```

2. Write a program that takes a positive integer $n > 100$ as input from the user and prints the factors for the numbers $4, \ldots, n$. For each number $4, \ldots, n$, the program should print all the factors except 1 and itself. If the number doesn't have any factor, the program should print that the number is prime. The program should also count the number of primes from 4 to $n$.

   Example output:

   ```
   Enter an integer > 100 : 120
   Factors for 4: 2
   Factors for 5: 5 is prime
   Factors for 6: 2 3
   ```

```
Factors for 7: 7 is prime
Factors for 8: 2 4
Factors for 9: 3
Factors for 10: 2 5
.................
The number of primes in 4 to 120 are : 28
```

3. Suppoe you are given a positive proper fraction $a/b$. Your problem is to write this fraction as

$$\frac{a}{b} = \frac{1}{n_1} + \frac{1}{n_2} + \ldots ... + \frac{1}{n_k}$$

where $n_1 < n_2 ... < n_k$ For example

$$\frac{4}{9} = \frac{1}{3} + \frac{1}{9}$$

$$\frac{11}{89} = \frac{1}{9} + \frac{1}{81} + \frac{1}{7209}$$

To obtain such a representation, you need to follow an iterative process. In each iteration, the fraction $a/b$ is decreased by $1/n$ for a suitable $n$, which is a positive integer satisfying

$$\frac{1}{n} \leq \frac{a}{b} \leq \frac{1}{n-1}$$

Repeat the process until the fraction $a/b$ reduces to zero. Write a program that reads two integers $a$, $b$ so that $a/b$ is a positive proper fraction. Apply the above iterative algorithm. During each iteration, report the value of $n$ for which $1/n$ is subtracted from the fraction.

Example output:

```
Supply a proper positive fraction.
Enter the numerator   : 11
Enter the denominator : 89
The sequence of values of n is: 9 81 7209
```