## Shiv Nadar University <u>CSD101: Introduction to Computing and Programming</u> Lab #3 Number based basic C programs

Max marks: 80. Due on/before:17.00, 11-Sep-2021.

- 1. (a) Write a program that reads two integers m and n (either or both could be negative) and prints the quotient and the remainder when m is divided by n. The remainder should be always positive. The division should not be done using built in  $\mathbf{C}$  operators but by repeated subtraction.
  - (b) Write a program that reads an integer n and prints out the number of even digits and odd digits in the number. For example, if n = 8793421 then the output should be Even=3, Odd=4, if n = 1000001 then Even=5, Odd=2

[15, 15=30]

- 2. This question is on prime numbers.
  - (a) A prime number is one that is divisible by 1 and itself. The smallest prime is 2. By convention 1 is not considered a prime. Write a program that reads in a number and outputs "Prime" if it is a prime and "Not Prime" if it is not.
  - (b) Using the above as a function write a program that reads a positive integer, say n, and prints out the first n primes.
  - (c) Twin primes are pairs of primes whose difference is 2. For example, the first few twin primes are:  $(3, 5), (5, 7), (11, 13), (17, 19), (29, 31) \dots$  The twin prime conjecture states that there are infinitely many such twin primes. Assuming this conjecture is true (it is yet to be proved) write a program to read a positive integer n and print the first n twin prime pairs.

[10, 10, 10=30]

[20]

3. Consider the problem of creating change for a certain amount of money by using the minimum number of coins/notes. So, assuming we have coins/notes of the following denominations 1, 2, 5, 10, 20, 50, 100, 200, 500, 2000 (as we do in India) write a program that reads in a positive number as amount and then outputs the minimum number of notes/coins and their denominations that will make up the amount.

So, if we had to make up the amount 63, we have 63 = 50 + 10 + 2 + 1 and the output should be: Number: 4

Denominations: 50=1 10=1 2=1 1=1

Denominations: 2000=1 500=2 200=1 50=1 20=2 5=1

Think about why you believe your program uses the minimum number of notes/coins.

L

4-Sep-2021