DSA Lab Assignment Set – I

1. Write a program to compute the factorial of an integer *n* iteratively and recursively. Check when there is overflow in the result and change the data types for accommodating higher values of inputs.

2. Write a program to generate the *nth* Fibonacci number iteratively and recursively. Check when there is overflow in the result and change the data types for accommodating higher values of inputs. Plot the Fibonacci number vs *n* graph.

3. Write programs for linear search and binary search for searching integers, floating point numbers and words in arrays of respective types.

4. Write a program to generate 1,00,000 random integers between 1 and 1,00,000 without repetitions and store them in a file in character mode one number per line. Study and use the functions in C related to random numbers.

5. Write a program to generate 1,00,000 random strings of capital letters of length 10 each, without repetitions and store them in a file in character mode one string per line.

6. Store the names of your classmates according to roll numbers in a text file one name per line. Write a program to find out from the file, the smallest and largest names and their lengths in number of characters. Write a function to sort the names alphabetically and store in a second file.

7. Take a four-digit prime number *P*. Generate a series of large integers *L* and for each member *Li* compute the remainder *Ri* after dividing *Li* by *P*. Tabulate *Li* and *Ri*. Repeat for seven other four digit prime numbers keeping *Li* fixed.

8. Convert your Name and Surname into large integers by juxtaposing integer ASCII codes for alphabet. Print the corresponding converted integer. Cut the large integers into two halves and add the two halves. Compute the remainder after dividing the result by the prime numbers *P* in problem 7.

9. Write a program to check whether a given number is perfect. If it is a perfect number, add '97' to each of its digits. Now consider the result as an ASCII series. Convert the same into equivalent character sequence.

Perfect Number is a Positive Integer (Greater Than Zero) whose divisor's sum is equal to the number
The Smallest perfect Number is 6

  Divisors of 6 are 1 ,2 and 3. The sum of its Divisors is 1 + 2 +3 =6 .

10. Write a program to find the maximum number of pieces that can be formed by making N cuts, where N is a positive integer.

11. Given a string and a string dictionary, write a program to find the longest string in the dictionary that can be formed by deleting some characters of the given string. If there are more than one possible results, return the longest word. If there is no possible result, return the empty string.

Example:

Input: d = {"ale", "apple", "monkey", "plea"}

 S = "abpcplea"

Output: "apple"

Explanation: After deleting "b", "c" "a" S became "apple" which is present in dictionary d.

12. Given two strings (*s* and *t*) of lowercase alphabets and a value *M*, write a program to detect if two strings are *M-anagrams* of each other or not.

Two strings are called M-anagrams if the below conditions are true.
 1. Both have same number of characters.
 2. Two strings can become anagram by changing at most M characters in a string.

If the result is true, further find the minimum number of operations required to convert *s* to *t*.

The list of permitted operations are:

- Insert a character at any position of the string.
- Remove any character from the string.
- Replace any character from the string with any other character.
- Shift any character to it's left or right position