National Institute of Technology Calicut
Department of Computer Science and Engineering
B. Tech. (CSE) – Third Semester
CS2092D: Programming Laboratory
Assignment –1

Submission deadline (on or before):
07th August 2019, 10:00:00 PM

Policies for Submission and Evaluation
You must submit your assignment in the moodle (Eduserver) course page, on or before the submission deadline. *Also, ensure that your programs in the assignment must compile and execute without errors in Athena server*. During evaluation your uploaded programs will be checked in Athena server only. Failure to execute programs in the assignment without compilation errors may lead to zero marks for that program.
Your submission will also be tested for plagiarism, by automated tools. In case your code fails to pass the test, you will be straightaway awarded zero marks for this assignment and considered by the examiner for awarding F grade in the course. Detection of ANY malpractice regarding the lab course will also lead to awarding an F grade.

Naming Conventions for Submission
Submit a single ZIP (.zip) file (do not submit in any other archived formats like .rar, .tar, .gz). The name of this file must be ASSG<NUMBER>_<ROLLNO>_<FIRST-NAME>.zip (For example: ASSG1_BxxyyyyCS_LAXMAN.zip). DO NOT add any other files (like temporary files, input files, etc.) except your source code, into the zip archive. The source codes must be named as
ASSG<NUMBER>_<ROLLNO>_<FIRST-NAME>_<PROGRAM-NUMBER>.<extension>
(For example: ASSG1_BxxyyyyCS_LAXMAN_1.c).
If you do not conform to the above naming conventions, your submission might not be recognized by some automated tools, and hence will lead to a score of 0 marks for the submission. So, make sure that you follow the naming conventions.

Standard of Conduct
Violations of academic integrity will be severely penalized. Each student is expected to adhere to high standards of ethical conduct, especially those related to cheating and plagiarism. Any submitted work MUST BE an individual effort. Any academic dishonesty will result in zero marks in the corresponding exam or evaluation and will be reported to the department council for record keeping and for permission to assign F grade in the course. The department policy on academic integrity can be found at: http://cse.nitc.ac.in/sites/default/files/Academic-Integrity.pdf.

General Instructions
Programs should be written in C language and compiled using C compiler in Linux platform. Invalid input should be detected and suitable error messages should be generated. Sample inputs are just indicative. Please do the programs in your free time either from System Software Lab (SSL) / Network Systems Lab (NSL), when the lab is not used for regular lab hours or do the programs using your own computer. Even if the programs work in your own computer, there is a chance that they may not work properly in the computers in SSL / NSL, due to some compatibility issues of the C compiler or the machine. Hence, before the evaluation day, check that your programs are ready for execution in the computers in NSL/SSL. Evaluation of few random questions from the following questions will be conducted on **08, August 2019**(Thursday).

Questions

1. Write a C program that reads an array of n integers and finds the sum of the contiguous subarray, which has the largest sum in the array. Print the largest sum, start index and end index of the contiguous subarray. Contiguous subarray is a continuous series of elements in the array.

> **Input**: Size of the array(n) 8
> **Array Elements** 10, -2, 15, 9, -8, 12, 20, -5
> **Output**: The Largest sum is 56 and the subset index begins at 0 and ends at 6

> **Input**: Size of the array(n) 6
> **Array Elements** 3, 34, -4, 12, -5, 2
> **Output**: The Largest sum is 45 and the subset index begins at 0 and ends at 3

2. Write a C program to circular shift an array of integers of size n from left to right by k elements. If k>n, then take k=k%n. Write a function *rotate(int arr[],int n,int k)* which return circular shifted array, where *n* is the size of the array and *k* is the number of times the array has to be shifted.

> **Input:** n= 5, k=2 , {1,2,3,4,5}
> **Output:** {4,5,1,2,3}

3. Write a C program to print square root of a number without using the library function *sqrt().* Print up   to 3 decimal places.

> **Input:** 30
> **Output**: 5.477

 4. Write a C program to find whether a **majority element** exists in an integer array. If it exists in an array, print it. A **majority element** in an array arr[] of size n is an element that appears more than n/2 times (and hence there is at most one such element).
> **Input**: {1,2,2,2,5}
> **Output**: 2

> **Input**: {1,2,2,4,5}
> **Output**: -1 //No majority element

5. Given a 2D matrix of size m x n, a man is at index (0,0) and has to reach index (m-1,n-1). Write a function *matrixTraverse(int arr[][],int m, int n)*, where *m* is no. of rows and *n* is no.of columns, to print the number of possible ways to reach (m-1,n-1) from (0,0). The only possible moves from (i,j) are
- (i+1,j)
- (i,j+1)
- (i+1,j+1) ,i.e., from a cell, we can either go right, down or down right.

> **Input**: m=3, n=2
> **Output**: 5

6. Given an array of integers, write a function ***spiralPrint (int arr[][],int m, int n)***, where ***m*** is no. of rows and ***n*** is no.of columns, to print a given matrix in spiral form.

> **Input**: 1 2 3
>   4 5 6
>   7 8 9

> **Output**: 1 2 3 6 9 8 7 4 5

7. Write a C program to convert a given binary number (base 2) to its equivalent decimal number (base 10) using function. For example, the decimal number corresponding to the binary number $(1011)_2$ is $1*2^0 + 1*2^1 + 0*2^2 + 1*2^3 = 11$

> **Input:** Enter a binary number: 110110111
> **Output:** The equivalent decimal number: 439

8. Write a program to check whether a given integer number is a perfect number or not, using functions.
Perfect Number is a positive integer whose value is equal to the sum of it's proper positive divisors excluding the number itself.
For e.g.: Number 6 is a Perfect Number where $6 = 1 + 2 + 3$,
       Number 28 is a Perfect Number where $28 = 1 + 2 + 4 + 7 + 14$.

> **Input 1:** Enter an integer number: 496
> **Output 1:** 496 is a Perfect Number.

> **Input 2:** Enter an integer number:695
> **Output 2**: 695 is not a Perfect Number.

9. Write a program to read an array of '**n**' integers. Find the sum of all four-digits even numbers in the given array using function (*Call by reference*).
> **Input:** Enter number of elements :8
>   Enter the elements : 109, 2015, 8, 423, 87, 927, 7618, 1212
> **Output:** The sum is : 7618 + 1212 = 8830

10. Write a program that reads an array of '**n**' elements containing only **0's** and **1's**. Your program should find the position of a 0 and replace it with a 1 to get the longest continuous sequence of 1's. Let this position of 0 be called p. Print p if such a 0 exists and print -1 if the original array contains only 1's. Assume the array indexing starts from 0.

> **Input 1**: Number of elements in the array: 13
> **Elements**: 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1
> **Output 1**: Position 9
> **Input 2**: Number of elements in the array :5
> **Elements**: 1, 1, 1, 1, 0
> **Output 2**: Position 4

11. Create an array of structures that stores the following details for each employee:

 NAME
 SALARY
 WORK PER DAY (in hours)

Assume that the number of employees is 10. Increase the salary depending on the number of hours of work per day as follows :

Work per day (in hours)	8	10	>10
Increase in salary	Rs.50	Rs.100	Rs.150

Write a menu driven program that uses the structure mentioned above and allows the user to perform the following operations:

1. Add an employee record
2. Display the details of all the employees who did not get any increment in salary
3. Display the details of all the employees with their final salaries.
4. Display the details of all the employees, given work per day(in hours).

Sample Input and Output

 (1) Add an employee record
 (2) Display the details of all the employees who did not get any increment in salary.
 (3) Display the details of all the employees with their final salaries.
 (4) Display the details of all the employees, given work per day(in hours).
 (5) Exit

 Input 1: Enter your choice: 1
 Name: Alice
 Salary: Rs.63,000
 Work per day (in hours): 8

 Input 2: Enter your choice: 1
 Name: Bob
 Salary: Rs.63,000
 Work per day (in hours): 10

 Input 3: Enter your choice: 1
 Name: Jack
 Salary: Rs.63,000
 Work per day (in hours): 6

 Input 4: Enter your choice: 2
 Output 4:
 Name   Final Salary
 Jack   Rs.63,000

Input 5: Enter your choice: 3
Output 5:

Name	Final Salary
Alice	Rs.63,050
Bob	Rs.63,100
Jack	Rs.63,000

Input 6: Enter your choice: 4
Enter the work per day(in hours): 8
Output 6:

Name	Final Salary
Alice	Rs.63,050

Input Enter your choice: 5

12. Write a menu-driven program which contains an array of function pointers. In the array each function pointer pointing to some function. Assume function pointer array contains 3 function pointer, which points to add(a,b), sub(a,b) and mul(a,b) functions.

add(a,b): Addition of two numbers.
sub(a,b): Subtraction of two numbers.
mul(a,b): Multiplication of two numbers.

Menu driven program should allows the user to perform the following operation:
(a) Addition of two numbers.
(b) Subtraction of two numbers.
(c) Multiplication of two numbers.

Sample Input and Output

(1) Addition of two numbers.
(2) Subtraction of two numbers.
(3) Multiplication of two numbers.
(4) Exit

Input : Enter your choice: 1
      Enter the value of a and b: 1   2
Output: 3
Input: Enter your choice: 2
      Enter the value of a and b: 2   2
Output: 0
Input: Enter your choice: 3
      Enter the value of a and b: 2   3
Output: 6
Input: Enter your choice: 4

13. Write a program to create the menu of a library. Create a structure containing book information like accession number, name of author, book title and flag to know whether the book is issued or not.

Create a menu in which the following can be done.

1 - Display book information

2 - Add a new book

3 - Display all the books in the library of a particular author

4 - Display the number of books of a particular title

5 - Display the total number of books in the library

6 - Issue a book

(If we issue a book, then its number gets decreased by 1 and if we add a book,
its number gets increased by 1)

Sample Input Output

1- Add a new book

2- Display book information

3- Display all the books in the library of a particular author

4- Display the number of books of a particular title

5-Display the total number of books in the library

6- Issue a book

7- Exit

Input: Select an operation from the menu listed above: 1

Enter details of book 1: Accession number: B001

Name of author: ABC

Book title: Digital system design

Issued: Not issued

Input: Select an operation from the menu listed above: 1

Enter details of book 2: Accession number: B002

Name of author: XYZ

Book title: Software engineering

Issued: Not issued

Input: Select an operation from the menu listed above: 1

Enter details of book 3: Accession number: B003

Name of author: PQR

Book title: Fundamentals of programming

Issued: Not issued

Input: Select an operation from the menu listed above: 2

Enter the title of book: Software engineering

Output: Accession number: B002

Name of author: XYZ

Issued: Not issued

Input: Select an operation from the menu listed above: 1

Enter details of book 3: Accession number: B004

Name of author: ABC

Book title: Computer architecture

Issued: Not issued

Input: Select an operation from the menu listed above: 3
    Enter the name of author: ABC
Output: Accession number: B001
    Name of author: ABC
    Book title: Digital system design
    Issued: Not issued

    Accession number: B004
    Name of author: ABC
    Book title: Computer architecture
    Issued: Not issued

Input: Select an operation from the menu listed above: 4
    Enter title of book: Fundamentals of programming
Output: 1

Input: Select an operation from the menu listed above: 5
Output: 4

Input: Select an operation from the menu listed above: 6
    Enter the accession number of book to be issuing: B001
                    Name of author: ABC
                    Book title: Digital system design

Input: Select an operation from the menu listed above: 4
    Enter title of book: Digital system design
Output: Number of books in this title is: 0

14. Write a C program to reverse a string using recursion. (Use pointers)
    Input: hello world
    Output: dlrow olleh

15. Write a C program to check whether a substring is present in the given string.
    Input 1: Enter the string : Good Morning
        Enter the substring : Morning
    Output 1: The substring Morning is present

    Input 2: Enter the string : NIT calicut
        Enter the substring : Kozhikode
    Output 2: The substring Kozhikode is not present

                    ****