**Assignment, Winter 2023**

**CSE 1102: Structured Programming Language Sessional**

**Instructors: Md. Mamun Hossain, Assistant Professor Problem Set 01**

**Problem Set 01**

Please write your solutions in A4/Letter size paper and provide the handwritten or printed hardcopy in the department. Aim for concise solutions; convoluted and obtuse descriptions might receive low marks, even when they are correct. Direct copy from internet or friends may produce an F grade in this course which may force you to retake the course in the next semester.

This Lab Report is meant to be an evaluation of your individual understanding, analyzing and programming capability into the course and should be completed without collaboration or outside help.

**Submission Deadline: 15 May, 2023**

**(Answer all the questions.)**

**Section A [150 points]: Problems - Solutions**

1. A cashier has currency notes of denominations 1, 5, 10, 50 and 100. If the amount to be withdrawn is input through the keyboard, find the total number of currency notes of each denomination the cashier will have to give to the with-drawer making sure the cashier hands out as many less number of notes as possible
2. An integer is entered through the keyboard. Write a function to obtain the prime factors of this number. For example, prime factors of 24 are 2, 2, 2 and 3 whereas prime factor of 35 are 5 and 7.
3. An integer is entered through keyboard. Write a recursive function to calculate the sum of the digits of the number entered. For example, if you take input 235, the output will be 10.
4. Write a C program to find the Greatest Common Divisor (GCD) and Least Common Divisor (LCM) of two positive integers.
5. Write a C program to take a letter from English alphabet as input and display both the previous and the next letters with ASCII codes. Assume that input will always be chosen from B to Y or b to y.

|  |  |
| --- | --- |
| Sample Input | Sample Output |
| Enter a letter: d | Previous letter with ASCII: c 99  Next letter with ASCII: e 101 |
| Enter a letter: B | Previous letter with ASCII: A 65  Next letter with ASCII: C 67 |

1. Write a program to input an integer through the keyboard until the user chooses to quit upon the appearance of options. Every time a number is entered. The program should display whether it is greater than, less than or equal to the previous integer. [Assume initial integer value is 41]

|  |
| --- |
| **Sample Input and output** |
| Enter an integer: 23  It is less than 41.  Do you want to continue (y/n)? y  Enter an integer: 32  It is greater than 23.  Do you want to continue (y/n)? y  Enter an integer: 32  It is equal to 32.  Do you want to continue (y/n)? n |

1. A Strong number is a number in which the sum of factorial of individual digits of the number is equal to the original number. For example, 145 is a Strong number because 1! + 4! + 5! = 145. Now, write a program to check whether the number is Strong number or not.
2. A Perfect number is a positive integer that is equal to the sum of all its proper positive divisors excluding the number itself. For example, 6 is a perfect number because the proper positive divisors of 6 are 1, 2, and 3 (excluding 6) and sum of all its proper divisors i.e. 1 + 2 + 3 = 6. Now, write a C program to check whether the number is Perfect number or not.
3. Two integers are entered through the keyboard which are stored in the variables base and exp; base and exp cannot be 0 at the same time. If this rule is disregarded, ask the user to re-enter the values for base and exp until the rule is obeyed. Write a program to find baseexp. You are NOT allowed to use the pow function in math.h library file. Your code will not be tested for large values of base and exponent as the result might exceed the range of an integer.
4. Write a C program to take a binary number as user input and determine its corresponding decimal number using function. The prototype of the function must be int binaryToDecimal(unsigned long long int).

1. The Fibonacci sequence is a sequence where the next term is the sum of the previous two terms. The first two terms of the Fibonacci sequence are 0 followed by 1. Now, write a recursive function to display the Fibonacci sequence of first n numbers where n is a positive integer less than 100.
2. If a string is unchanged while reading from both end, it is said to be a palindrome. Now, write a C program to determine whether a string is palindrome or not. Don’t use any built in string functions.
3. Using pointers, write a function named xstrcat() which takes 2 strings as inputs, concatenates them and outputs the concatenated string. For example, if string1 is BAUST and string2 is CSE, the concatenated string is BAUSTCSE. Print the results. Don’t use string.h header file.
4. Ask the user to enter a sentence which is not more than 50 characters long. Write a function delVowel() which deletes all the vowels in the string. Do this task using pointer operations only. Print both the original and final string in main(). Don’t use string.h header file.
5. Create a 1D array dynamically. Incorporate the following specifications.

* Send the base address of the array and an integer to a function called sort().
* The integer is a user input which is either 0 or 1. If the user does not enter either 0 or 1, keeping asking the user until you get the right input.
* If the user input is 0, sort the array in the function in increasing order.
* If the user input is 1, sort the array in the function in decreasing order.
* Print the sorted array in main().

Then apply binary search algorithm to find if the user entered number is found in the array. If found, exit the search operation with an appropriate message. If you do not find the number, display an appropriate message.

1. Suppose, XYZ wants to make a list of integers dynamically. Then he will perform the insertion and deletion operations at a desired location in the list as his wish. Write a C program to help him solve the problem. You must create individual functions for both operations.

|  |
| --- |
| **Sample Input and Output** |
| How many elements? 5  Enter the elements: 4 7 1 9 2  Which operation do you want to perform now? (1. Insertion or 2. Deletion)  Enter your choice: 1  Enter the element you want to insert: 23  At which position? 3  After insertion, the elements are: 4 7 23 1 9 2  Which operation do you want to perform now? (1. Insertion or 2. Deletion)  Enter your choice: 2  Enter the element you want to delete: 7  After deletion, the elements are: 4 23 1 9 2  Which operation do you want to perform now? (1. Insertion or 2. Deletion)  Enter your choice: 3  Sorry! Invalid choice. |

1. Write a program to take as input height of several students in centimeters. Read from console the number of students (integer), then as many real values. You must allocate memory dynamically. After storing the heights, find and print the height difference between the tallest and the shortest student. Then free the allocated memory.
2. Write a C program to read two matrices from user and then perform the multiplication operation of these matrices and finally display the result.
3. Write a program to read a matrix and display the sum of the diagonal elements of the matrix.
4. Write a program to swap the values of two integer variables with call by reference.
5. Using a pointer to pointer and appropriate dynamic memory allocation, write a program to store a matrix of integers and then print the average of the integers in each column of the matrix. The input starts with 2 integers representing the number of rows (r) and columns (c) respectively. Then the numbers in the matrix are given in a row major order (i.e. the first c integers represent the first row, the next c integers represent the second row and so on). Before the program exits, you must appropriately free the dynamically allocated memory.
6. Write a recursive function to find and return the largest element in an array of integers. Do not use loops, static or global variables. The prototype of the function must be: ***int largest (int x []), int n);*** Here x is the array of integers, while n represents the number of integers in the array. After implementing the function, write a simple main function to demonstrate how you would make the initial call to the recursive function.
7. Create a structure data type to specify data on students as follows: ID Number, Name, Department, Course and Year of joining. Assume there are 5 students in the class. Hardcode the student data with ID as a 1 digit number (1 to 5), Student Name as a string of your choice, Department as a string of your choice, Course as a number of your choice and Year of Joining as either 2022 or 2023.

* Write a function to print names of all students who joined in a particular year. User gives the year as input during runtime.
* Write a function to print student data based on the ID number entered by the user.

1. Write a C program to add two distances in feet and inches using union.
2. Write a C program to create and store information in a file, then read the file and count the number of words and characters in the file.

**Rubric**:

* -1 point for each wrong answer
* 1 points for indentation, 1 points for documentation, 1 points for coherence, 2 points for logical integrity
* 5 points for correct answer
* Partial credit may be awarded if there is work shown of a correct approach that does not yield the correct solution.