Lab No.1 Structured Programming Refresher

1.1 Objectives of the lab

Reinstate the concepts of structured programming such as
1. decision making
2. iterations
3. recursion
4. functions
5. structures and arrays
6. pointers
7. file handling

1.2 In-Lab

1.2.1 Activity

Write a function that reverses elements stored in an array

Sample OUTPUT of the function

Enter length of the array: 5

Enter the elements of the array: 11   22   33   44   55

Input array is: 11   22   33   44   55

Reverse of the input array is: 55   44   33   22   11

Note: Actual input array elements are to be swapped i.e. in the above example, swap array element 0 with element 4 and so forth. Temporary array for reversing the elements is not allowed.

1.2.2 Activity

Write a program that computes and prints both the mean and the standard deviation of the input data.

The standard deviation of the n numbers ao, . . ., an, is defined by the formula:

$$\sigma = \sqrt{\frac{\sum_{i=0}^{n-1}(a_i - \mu)^2}{(n-1)}}$$

where μ is the mean of the data. This means: square each deviation a[i] - mean; sum those squares; divide the sum by n-l; and take the square root finally.

1.2.3 Activity

Write a program to find transpose of an nxn matrix.

1.2.4 Activity

The greatest common divisor of integers x and y is the largest integer that evenly divides both x and y. Write a recursive function gcd that returns the greatest common divisor of x and y. The gcd of x and y is defined recursively as follows: If y is equal to 0, then gcd(x, y) is x; otherwise gcd(x,y) is gcd(y, x%y), where % is the modulus operator.

1.2.5 Activity

Write a program that reads a group of numbers from the user and places them in an array of type float. Once the numbers are stored in the array, the program should find the largest from entered numbers and print the result. Use pointer notation wherever possible.

1.3 Home-Lab

1.2.6 Activity

A company wants to transmit data over the telephone, but they are concerned that their phones may be tapped. All of their data are transmitted as four-digit integers. They have asked you to write a program that encrypts their data so that it can be transmitted more securely. Your program should read a four-digit integer and encrypt it as follows: Replace each digit by (sum of that digit plus 7) modulus 10. Then swap the first digit with the third, swap the second digit with the fourth and print the encrypted integer.

Write a separate program that inputs an encrypted four-digit integer and decrypts it to form the original number.

1.2.7 Activity

Input an integer containing only 0s and 1s (i.e. a "binary" integer) and prints its decimal equivalent.

(Hint: use the modulus and division operators to pick off the "binary" number's digits one at a time from right to left. Just as in the decimal number system where the rightmost digit has a positional value of 1 and the next digit left has a positional value of 10, then 100, then 1000 etc., in the binary number system, the rightmost digit has a positional value of 1, the next digit left has a positional value of 2, then 4, then 8, etc. Thus the decimal number 234 can be interpreted as 4*1+3*10+2*100. The decimal equivalent of binary 1101 is 1*1 + 0*2+1*4+1*8 or 1+0+4+8 or 13.

1.2.8 Activity

Write a program that would store one hundred random numbers, ranging from 0 to 99, in an array; and

displays first repeating number in the array.

Note: Use of nested loops is not allowed.

Hint: Numbers stored in the array has range [0, 99] and study the properties of % to figure out a way of getting a random number in range [0, 99].

1.2.9 Activity

Write a program that creates an array containing the following integer numbers 6, 4, 8, 9, 12, 15, 13, 16, 17, and 20. Write the data in the array to a text file using ofstream with each number separated by new line and read it into another array using ifstream. Write functions for both writing and reading purpose.

1.4 References:

1   **Object-Oriented Programming in C++ by** *Robert Lafore*

2   **How to Program C++ by** *Deitel & Deitel*