National University of Computer and Emerging Sciences



Assignment 01

For

Object Oriented Programming Lab

|  |  |
| --- | --- |
| Course Instructor(s) | Dr. Irfan ul Haq |
| Lab Instructor(s) | Mr. Mughees Ismail |
| Semester | Spring 2020 |

**FAST School of Computing**

# Instructions:

1. Make a word document with the naming convention “SECTION\_ LAB#\_ROLLNO” and put all your source code and snapshots of its output in it. Make sure your word file is formatted properly.
2. Plagiarism is strictly prohibited.
3. Do not discuss solutions with one another.

**Q1:**

Write a C++ program that Declare Array1 and Array2 of size 5 and Array3 of size 10.

Declare Three Pointers Named One, Two and Three that point to these arrays.

1. Take input from user in Array1 and Array2.

2. Sort Array1 and Array2.

3. Marge Array1 and Array2 and store it to Array3 in Sorted order.

4. Clear the screen and display Array1, Array2 and Array3.

5. Also display the even entries and odd entries of Array3 and number of even and odd entries in Array 3.

Note:

You can only declare arrays and pointers and calling functions in main(). You have to pass only pointers in function calling (Not Arrays).

**Q2:**

Write a C++ program that can store string in file and find data from file.

Main Menu:

1. STORE STRING IN FILE

2. FIND WORD FROM FILE

**STORE STRING IN FILE:**

Take input from user and store it to a file name “**PF\_LAB.txt**”.

And ask user to store another string or not.

Example: **I love my Pakistan.**

**FIND WORD FROM FILE:**

Take input from user find the word from file name “**PF\_LAB.txt**”.

Example Input: **MY**

Your program converts all the characters in **UPPERCASE** and choose word from file and convert the characters in **UPPERCASE** and check either equal or not.

If word exist in file Display a massage **The Word “MY" Found in file at position “3".**

If word not found Display massage **WORD “MY" not Found in file.**

**NOTE:**

You can only call functions in main().At the end of any Menu your program clear the screen and display main menu again.

**Q3:** Write a C++ program that read and write data in file.

1. Enter new Student

2. **Search a student**

3. Display all the students

**ENTER NEW STUDENT:**

User Enter the Roll no of Student

Write a program that check either this roll no exist already in file or.

If Roll no not found in file Take Student Name, Father Name, Section.

And store all the info in file name **"All\_Students\_info.txt".**

In such format: [Roll No][Student Name][Father Name][Section]

If Roll no exist already in file display massage: **ROLL NO Already exist.**

Example of Stored data in file:

18F-0213\*Abdul Ahad\*Waseem Akram\*B

18F-0277\*Arslan Ahmad\*Altaf Ahmad\*B

**SEARCH A STUDENT:**

User Enter the Roll No of a student and your program find the Roll No in file.

If Roll No not found in file Display: **Roll No Not Found.** And again take input Roll No.

If Roll No Found Display the Student Details.

Example:

INPUT: 18F-0213

OUTPUT:

Student Name: Abdul Ahad

Father Name: Waseem Akram

Section: B

**DISPLAY ALL STUDENTS:**

You have to display all the student’s information of all the student exist in file.

**NOTE:**

Display all the Error massages like **File is Empty.**

And you can call functions in main() only.

**Q4:**

Write a C++ Program which take a string as an input from user and check that either all the characters of that string belongs to single line of Keyboard.

If all the Characters belongs to single Keyboard line than Display the Line Number.

Example:

**INPUT: Roy**

**Output:** Roy Belongs to 2nd Line of Keyboard.

**INPUT: Lab**

**OUTPUT:** Lab Not belongs to any Single Line of Keyboard.

**Q5:**

Write a C++ program that print the Blinking Diamond of Size Given by the User.

Hint: Blink Diamond (print Hollow Diamond after a second clear screen and print Fill Diamond).

**Q6:**

Write a C++ Program that take two numbers for user as input and display choice.

1) LCM

2) HCF

3) GCD

4) POWER

5) Addition

6) Subtraction

7) Multiplication

8) Division

User input the Function and you have to run that function only using FUNCTIONS.

**Q7:**

****

****

**Q8:**

Write a program that will input a number and check if it is even or odd without using modulus (%) operator and a loop.

**Q9:**

Write a C++ Program to encode an image which is being input as a matrix.

Input a 3x3 matrix containing only 0s and 1s, find the Transpose of a 3x3 Matrix. Easy? Let’s invert the transposed matrix. If 1 1 0 are the entries, the inverted row will be 0 0 1. After inverting the whole matrix, use the last three digits of your roll number, add them and multiply them with the entries containing 1. I.e: In 18F-0213, add the last 3 digits 2 + 1 + 3 and multiply. Now, write a function decode which will return the original matrix again.

Note: Use function encode and decode, call these functions transpose/invert/multiply/divide for the sub tasks as instructed above.

Sample Output:

Clear screen.

Encoding:

Print original 3x3 matrix.

Print transposed matrix.

Print inverted matrix.

Print resultant matrix.

Decoding:

Print the above steps in reverse.

**Q10:**

A Stack is a **LIFO** structure works on the property of Last in First Out. This means whatever you have inserted at the last, you can only get back that last inserted value. You will make a character array which will work like a stack.

Your task is to input a string from the user and reverse it using a stack.

At first, you will create a character array for the size of string entered by the user. After doing so, you’ll insert the elements in to the character array and then starting from the end of the character array you will fetch all the entries one by one and store it back to the string. Your resultant string will be reversed.

Sample Input: Hello World.

Sample Output: .dlroW olleH

**Q11:**

Write a C++ function to Perform Matrix Multiplication.

**Q12:**

When a message is transmitted in secret code over a transmission channel, it is usually

sent as a sequence of bits, that is, 0s and 1s. Due to noise in the transmission channel,

the transmitted message may become corrupted. That is, the message received at the

destination is not the same as the message transmitted; some of the bits may have been

changed. There are several techniques to check the validity of the transmitted message

at the destination. One technique is to transmit the same message twice. At the

destination, both copies of the message are compared bit by bit. If the corresponding bits

are the same, the message received is error-free.

Let’s write a program to check whether the message received at the destination is error-

free. For simplicity, assume that the secret code representing the message is a sequence

of digits (0 to 9) and the maximum length of the message is 250 digits.

Also, the first number in the message is the length of the message. For example, if the

secret code is:

7 9 2 7 8 3 5 6

then the actual message is 7 digits long, and it is transmitted twice.

The above message is transmitted as:

7 9 2 7 8 3 5 6 7 9 2 7 8 3 5 6

Input Sequence of numbers containing the secret code and its copy

Output The secret code, its copy, and a message—if the received code is error-

free—in the following form:

Code Digit Code Digit Copy

9 9

2 2

7 7

8 8

3 3

5 5

6 6

Message transmitted OK.

Because we have to compare the corresponding digits of the secret code and its copy,

we first read the secret code and store it in an array. Then we read the first digit of the

copy and compare it with the first digit of the secret code, and so on. If any of the

corresponding digits are not the same, we indicate this fact by printing a message next to

the digits. Because the maximum length of the message is 250, we use an array of 250

components. The first number in the secret code, and in the copy of the secret code,

indicates the length of the code.

Keep your main as simple as possible. Do everything else in the functions.

**Q13:**

A small airline has just purchased a computer for its new automated reservations system.

You’ve been asked to program the new system. You are to write a program to assign

seats on each flight of the airline’s only plane (capacity: 100 seats – 50 in first class and

50 in economy class).

Your program should display the following menu of alternatives—Please type 1 for "First

Class" and Please type 2 for "Economy". If the person types 1, your program should

assign a seat in the first-class section (seats 1–50). If the person types 2, your program

should assign a seat in the economy section (seats 51–100). Your program should print

a boarding pass indicating the person’s seat number and whether it’s in the first class or

economy section of the plane.

Use a two-dimensional array to represent the seating chart of the plane. Initialize all the

elements of the array to false to indicate that all seats are empty. As each seat is

assigned, set the corresponding elements of the array to true to indicate that the seat is

no longer available. Your program should, of course, never assign a seat that has already

been assigned. When the first-class section is full, your program should ask the person if

it’s acceptable to be placed in the economy section (and vice versa). If yes, then make

the appropriate seat assignment. If no, then print the message "Next flight leaves in 3

hours."

The seating of the plane is as following.

Each row contains 10 seats. Seats 1 through 50 are first class and 51 through 100 are

economy.

The pass should contain proper indentation (use setw(),setfill() etc. from the iomanip

library).

**Q14:**

Given an array of strings, sort the array using Selection Sort.

Examples:

Input : paper true soap floppy flower

Output : floppy, flower, paper, soap, true

**Q15:**

Write a function equalsIgnoreCase, which receives two char arrays and their sizes, and returns true

if the two char arrays contain the same characters irrespective of the case. For example, for

character arrays {'a', 'B', 'c'} and {'A', 'b', 'c'}, the function returns true, but for {'a', 'B', 'c'} and {'a',

B'}, or {'a', 'B', 'c'} and {'X', 'b', 'z'}, the function returns false.

**Q16:**

Write a function CountDigit which receives two arguments: a char array and the size of the array

(of type int). This function counts the number of digit letters in the char array, and returns the

count (of type int). Use the C++ predefined function isdigit (defined in <cctype> ).\

**Q17:**

Write a program which will be given as input an array and an integer p. The

program will then cyclically shift the array p positions to the right: each element is

moved p positions to the right, while the last p elements are moved to the

beginning of the array. For example: if we have the array [1 2 3 4 5 6], shifting 2

positions to the right should give the array [5 6 1 2 3 4]. Your function should

work correctly for negative values of p.

**Q18:**

Use a one-dimensional array to solve the following problem. Read 20 numbers, each

of which is between 25 and 75 by using rand ( ) function. As each number is read,

validate it and store it in the array only if it is not a duplicate of a number already

read. After reading all the values pass the array to function name display ( ) and

display all the values on console.

**Q19:**

Write a C++ program to find the number of pairs of integers in a given array of

integers whose sum is equal to a specified number.

**Q20:**

Given an integer n and a prime number p, find the largest x such that p^x

(p raised to power x) divides n! (factorial)

Examples:

Input: n = 7, p = 3

Output: x = 2

3^2 divides 7! and 2 is the largest such power of 3.

Input: n = 10, p = 3

Output: x = 4

3^4 divides 10! and 4 is the largest such power of 3.

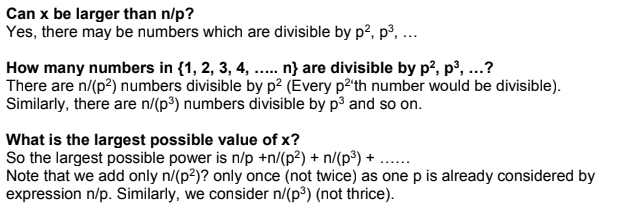
n! is multiplication of {1, 2, 3, 4, ...n}.

How many numbers in {1, 2, 3, 4, ..... n} are divisible by p?

Every p’th number is divisible by p in {1, 2, 3, 4, .... n}. Therefore in n!, there are n/p

numbers divisible by p. So we know that the value of x (largest power of p that divides n!)

is at-least n/p.



Use pass by reference only in the functions that you make.

**Q21:**

Make a void function which takes two double parameter and finds the equation of straight

line i.e.

Void line(double &x0, double &y0)

**Q22:**

An integer is said to be a perfect number if the sum of its divisors, including 1 (but not the

number itself), is equal to the number. For example, 6 is a perfect number, because 6 =

1 + 2 + 3. Write a function isPerfect(int no) that determines whether parameter number is

a perfect number. Use this function in a program that determines and prints all the perfect

numbers between 1 and 1000. Print the divisors of each perfect number to confirm that

the number is indeed perfect.

**Q23:**

Create two matrices of user defined size dynamically. Perform the following operations on matrices.

• Addition

• Subtraction

• Multiplication

For each operation write a separate function.