Object Oriented Programming Lab 2019 **Lab14: File Handling**

Instructions

- There are no test test cases for this particular lab.
- kindly follow the instructions that are written in each question. Failing to follow the instructions can lead in reduction of marks.
- In case of any query, please raise your hands and we will be there to solve your query.
- Please concentrate, understand, and code. Good Luck :)

Task 1

Create a program for writing records of 10 students in a text fie Students.txt. Store roll no, first name, last name, department and section of student.

Output should be like this in the file.

Table 1: Student.txt

123 Asad Ali CS A.
321 Abbas Khan EE C.
342 Bilal Haider FSM A.

Task 2

Now write another program that reads records of all students in a text fie Students.txt. Program must read roll no, first name, last name, department and section of student. Program must display the read information from the fie.

Task 3

Design a program for reading two text files, f1.txt and f2.txt and identifying the difference between the lines of both text files, i.e. writing on the screen the lines that differ in both files, adding ¡ if the line corresponds to f1.txt, and ¿ if it corresponds to f2.txt. You can consider getline function of istream class.

f1.txt	f2.txt
This is a test line. Hello World.	This is a test line. Hello Guys.

Output:
< Hello World.
> Hello Guys.

Task 4

Write a function called *printLines()* receiving two arguments: the first one must be a positive integer n, and the second the name of a text file as string. The function must print on the screen the last n lines of the given file, if there are less number of lines then n they must be printed as well.

Task 5

Write a program to implement a telephone directory using formatted I/O file. You should write a class **TeleDirectory** for storing the name, number and address of telephone holder. Your program should be capable of writing name, telephone number and address into a data file *teledir.txt*. Then write class functions *searchByName*: which should read the file and must return the Phone number of searched person and *searchByNumber*: which should read the file and must return the name of person holding the searched phone number. Also write a class function *printDirectory* which must read the complete directory and print all the records of the directory.

Task 6

Write a function named *arrayToFile*. The function should accept three arguments: the name of a file, a pointer to an int array, and the size of the array. The function should open the specified file in binary mode and write the contents of the array along with its size to the file, and then close the file.

Write another function named *fileToArray*. This function should accept two arguments: the name of a file, a pointer to an int array. The function should open the specified file in binary mode, read the size of array and should allocate memory to the pointer and then read the contents of the array, and then close the file. This function should return the size of array read from the file.

Write a complete program that demonstrates these functions by using the *arrayToFile* function to write an array to the file, and then using the *fileToArray* function to read the data from the same file

After the data is read from the file into the array, display the arrays contents on the screen.

Task 7

Lets re-implement the **Student** Class from Question 1 & 2. Now Write a class Student with the following data members:

```
class Student
{
    char name[50];
    int age;
    char phone[24];
    float cgpa;
    int rollNo;
};
```

Write getters and setters of given data members in camel casing Notation.

Now write class member function *void WriteToFile(ofstream &)* which takes ofstream object as argument and writes the students object data in binary format in the opened file. You are required to write the data of object member by member. Also before sending the ofstream object as argument the file must be opened with the name *StudentData*

Now write class member function *void ReadFromFile(ifstream &)* which takes ifstream object as argument and student object data in binary format from the opened file and fill the student information.

Finally create main function and create an array of 5 students write their data into the file name *StudentData* and then read it from the file into a separate array of 5 students.