



Programming Fundamentals

Lab Manual - Week 13

Introduction

Welcome Back to your favorite Programming Lab students. In this lab manual, we shall work together to learn and implement new programming concepts.

Skill: Learning fstream to Read, Write and Append in files

Let's do some coding.

Introduction

We have learned how to create programs that store user-entered data and can use this data to carry out different tasks. However, a key limitation was that all the information stored in various variables was lost once the program was closed. Therefore, we were unable to store the data permanently on the computer. This week, we will discover how to store data permanently on the computer.

Task 01(WP): Write a program that writes “this is sample text” to a file.

Step 01: Include the fstream library in your program

```
# include <fstream>
```

Step 02: Create a file variable in your program

```
fstream file;
```

Step 03: Open the file in desired mode using the variable created in the previous step

```
file.open(“example.txt”, ios::out);
```

We can open the file in one of the following modes

	Description
ios::in	We use the mode when we want to read from the file

ios::out	We use the mode when we want to write to the file
ios::app	We use the mode when we want to append data in the file

Step 04: Learning fstream to Read, Write and Append in files



Programming Fundamentals

Lab Manual - Week 13

Step 04: Write to the text using a variable or a string.

```
file << "This is sample text";
OR
string text = "This is sample text";
file << text;
```

Step 05: Close the file when you have performed your desired operations on the file.

```
file.close();
```

Task 02(OP):

- (a) Write a program that takes an integer from the user and writes to a file. (b) Write a program that takes a decimal input from the user and writes to a file. (c) Write a program that takes a character input from the user and writes to a file.

Hint: Remember the five steps to use a file.

Task 03(CL): Write a program that reads the name stored in the file and prints it on screen.

```

#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    string name;
    fstream file;
    file.open("example.txt", ios::in);
    file >> name;
    file.close();
    cout << "The name in file: " << name;
}

```

In this solution, we have **opened the file in the read mode** and stored the string from the file in the **name** variable.

Stop Learning fstream to Read, Write and Append in files



Programming Fundamentals



Lab Manual - Week 13

Task 04(OP):

- Write a program that reads an integer stored in the file and prints it on screen.
- Write a program that reads a decimal stored in the file and prints in on screen.
- Write a program that reads a character stored in the file and prints it on screen.

Hint: Remember the five steps to use a file.

Question:

Now, what if we want to read a complete line (with the spaces) from the file ? Can we use the same technique for reading from the file as we have learned earlier in the manual ?

Consider the following task for better understanding.

Task 05(WP): Write a program that reads a complete line stored in the file.

```

#include <iostream>
#include <fstream>
using namespace std;
int main()
{
    string line;
    fstream file;
    file.open("example.txt", ios::in);
    getline(file, line);
    file.close();
    cout << "The Line in file: " << endl << line;
}

```

In this solution, we have used the `getline(file, line)` function to take input from the file that will return a complete line in our variable.

Session 13 Learning fstream to Read, Write and Append in files



Programming Fundamentals

Lab Manual - Week 13

Consider that we want to read a file with multiple lines and we want to display all its contents on the console screen.

Task 06(WP): Write a c++ program that displays all the contents of a file.

```

#include <iostream>
#include <fstream>
using namespace std;
main()
{
    string line;
    fstream file;
    file.open("example.txt", ios::in);
    while(!file.eof())
    {
        getline(file, line);
        cout << line<<endl;
    }
    file.close();
}

```

Task 01(CP): Write a program that calls a function with a string ("Task1.txt") and then the function reads the text file named **Task1.txt** line by line and returns the total number of lines present in the file.

Function Prototype:

```
int countLines(string fileName)
```

Text File:

The sun peeked over the horizon,
painting the sky in shades of orange and pink.
Birds chirped cheerfully,
welcoming the new day.
Sarah sat on her porch,
sipping a cup of steaming coffee.
She took a deep breath,
filling her lungs with the crisp morning air,
and smiled as the world woke up around her.

Test Case:

```
G:\Semesters\Programming Fundamentals (Fall 2023)\Week 13\Lab Tasks>Task1.exe  
Total number of lines: 9
```

Skill: learning fstream to Read, Write and Append in files



Programming Fundamentals

Lab Manual - Week 13



Task 02(CP): Write a program that calls a function with a string (“Task2.txt”) and then the function reads the text file named **Task2.txt** line by line and returns the the total number of characters in the file.

Function Prototype:

```
int countCharacters(string fileName)
```

Text File:

The sun peeked over the horizon,
painting the sky in shades of orange and pink.
Birds chirped cheerfully,
welcoming the new day.
Sarah sat on her porch,
sipping a cup of steaming coffee.
She took a deep breath,
filling her lungs with the crisp morning air,
and smiled as the world woke up around her.

Test Case:

```
G:\Semesters\Programming Fundamentals (Fall 2023)\Week 13\Lab Tasks>Task2.exe  
Total number of characters: 292
```

Task 03(CP): You are tasked with creating a program in C++ that calls a function named `calculateFrequency` with the parameter of a filename and the function then reads that text file and determines the frequency of a specific character within that file. First line of the text file will contain the character whose frequency you have to calculate. After that all the next lines will contain the text until the end of the file. While calculating frequency, your code should not be case sensitive. i.e., `s` and `S` will both be counted while calculating the frequency for `s`.

Function Prototype:

```
int calculateFrequency(string fileName)
```

Text File:

s



ing fstream to Read, Write and Append in files



Programming Fundamentals

Lab Manual - Week 13

The sun peeked over the horizon,
painting the sky in shades of orange and pink.
Birds chirped cheerfully,
welcoming the new day.
Sarah sat on her porch,
sipping a cup of steaming coffee.
She took a deep breath,
filling her lungs with the crisp morning air,
and smiled as the world woke up around her.

Test Case:

Task 04(CP): You are assigned to develop a project in which the project manager wants the following functionalities.

- Create a function that will ask the student's details from the console. Keep taking the input until the user enters No.

Information contains:

1. Student Name
2. Student Age
3. Student Matric marks
4. Student Fsc marks
5. Student Ecat Marks

- Create a separate function to save all the entered information in the separate lines of the file **student.txt**.

Note: Do not make Global arrays.

Function Prototypes:

- `void getStudentDetails(string names[], int ages[], float matricMarks[], float fscMarks[], float ecatMarks[], int size, int &count)`
- `void saveToFile(string names[], int ages[], float matricMarks[], float fscMarks[], float ecatMarks[], int count, string fileName)`

Skill: learning fstream to Read, Write and Append in files



Programming Fundamentals

Lab Manual - Week 13



Test Case:

Input Output	

Task 05(CP):

Write a program that reads the data from the **Task5.txt** file and stores the data of students in the **topperStudents.txt**

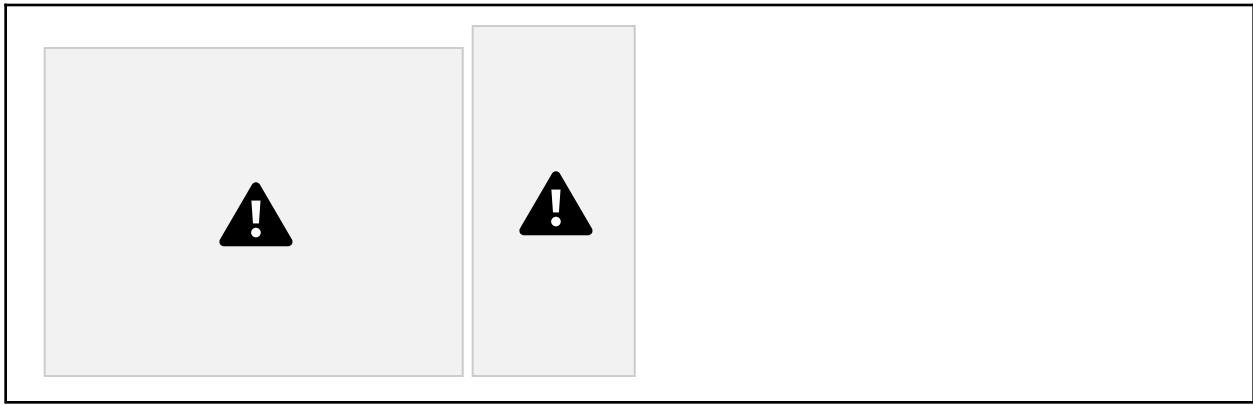
Information contains:

1. Student Name
2. Admission Number
3. Percentage

Instruction: Store the data of only those students having a percentage higher than 70.

Function Prototypes:

- `void getStudentDetails(string names[], int adNumbers[], float percentages[], int size, int &count, string fileName)`
- `void saveToFile(string names[], int adNumbers[], float percentages[], int count, string fileName)`



ing fstream to Read, Write and Append in files



Programming Fundamentals

Lab Manual - Week 13

Task 06(CP): Write a function in C++ to count the number of lines from a text file "Task6.txt" which is not starting with an alphabet given on the first line of the file.

Text File:

f
You're on a journey that's uniquely yours,
filled with challenges and triumphs that shape who you are.
Embrace the learning process,
for it's not just about grades but about the growth you experience along the
way. Believe in your abilities,
for within you lies incredible potential waiting to be unleashed.
Every setback is a lesson,
every hurdle a chance to grow stronger.
Keep your dreams vivid and your determination fierce.
You're crafting a future that's shaped by your dedication and hard
work. So,
stay curious,
stay passionate,
and keep reaching for the stars because your efforts today will pave the way
for a brighter tomorrow.
You've got this!

Function Prototype:

```
int countLines(string fileName)
```

Test Case:

Task 07(CP):

Write a function `displayWords(string fileName)` in C++ to read lines from a text file "Task7.txt", and display those words, which are less than 4 characters.

Text File:

You're on a journey that's uniquely yours,



ing fstream to Read, Write and Append in files



Programming Fundamentals

Lab Manual - Week 13

filled with challenges and triumphs that shape who you are.
Embrace the learning process,
for it's not just about grades but about the growth you experience along the way.
Believe in your abilities,
for within you lies incredible potential waiting to be unleashed.
Every setback is a lesson,
every hurdle a chance to grow stronger.
Keep your dreams vivid and your determination fierce.
You're crafting a future that's shaped by your dedication and hard work.
So,
stay curious,
stay passionate,
and keep reaching for the stars because your efforts today will pave the way for a
brighter tomorrow.
You've got this!

Console Output

Task 08(CP):

Develop a SignIn SignUp Application using File System for your Business Application.

- As a user, when I SignUp to the system the username, password and role are stored into the file.
- As a user, when I SignIn to the system the program should let me in if i am a valid user.

Store the username and password and role in the parallel array as well as the file named **users.txt**.

Read the file **users.txt** at the start of the application and read all the names, passwords and role in the parallel arrays

Good Luck and Best Wishes !!

Happy Coding ahead :)

Skill: learning fstream to Read, Write and Append in files