**Fundamentals of Programming**

**Lab Journal - Lab # 8**

Name: Mohammad Arsalan Shakil

Enrollment #: 01-134181-032

Class: BSCS-1A

**Objective**

This is lab will introduce students to arrays, array input/output and array operations.

**Open Function and Some of its Modes:**

In order to open a file with a stream object we use its member function open(): **open (filename, mode);**

Where filename is a null-terminated character sequence of type const char \* (the same type that string literals have) representing the name of the file to be opened, and mode is an optional parameter with a combination of the following flags: **ios::in** Open for input operations. **ios::out** Open for output operations.

**ios::binary** Open in binary mode.

**ios::ate** Set the initial position at the end of the file. If this flag is not set to any value, the initial position is the beginning of the file. **ios::app** All output operations are performed at the end of the file, appending the content to the current content of the file. This flag can only be used in streams open for output-only operations.

**ios::trunc** If the file opened for output operations already existed before, its previous content is deleted and replaced by the new one.

**Exercise 1**

Write a program that writes three characters in a text file.

**Code :**

#include<iostream>

#include<conio.h>

#include<fstream>

using namespace std;

int main()

{

cout << "Creating a File\n\n\n"; char x, y, z;

cout << "Enter 3 Char for output\n";

cin >> x >> y >> z;

ofstream my\_file;

my\_file.open("Arsalan.txt");

my\_file << x << y << z;

my\_file.close();

cout << "File is created";

\_getch();

return 0;

}

**Exercise 2**

Write a program that reads the characters from a text file. It counts total number of character and total number of vowels in the file.

**Code :**

#include<iostream>

#include<conio.h>

#include<fstream>

using namespace std;

int main()

{

char my\_char; int num\_line = 0, vowel = 0;

ifstream fin;

fin.open("my input file.txt");

while (!fin.eof())

{

fin.get(my\_char);

cout << my\_char;

if (my\_char == 'a' || my\_char == 'e' || my\_char == 'i' || my\_char == 'o' || my\_char == 'u')

{

vowel++;

}

if (my\_char == '\n')

{

num\_line++;

}

}

cout << "Number of vowels : " << vowel << endl << "Number of lines : " << num\_line;

\_getch();

return 0;

}

**Exercise 3**

Write a program that reads a text file which has a poem written in it. (Write down the poem yourself in a text file manually). Your program should display that poem and count total number of lines in the file.

**Code :**

#include<iostream>

#include<conio.h>

#include<fstream>

using namespace std;

int main()

{

char my\_char; int num\_line = 0, vowel = 0;

ifstream fin;

fin.open("Poem.txt");

while (!fin.eof())

{

fin.get(my\_char);

cout << my\_char;

if (my\_char == 'a' || my\_char == 'e' || my\_char == 'i' || my\_char == 'o' || my\_char == 'u')

{

vowel++;

}

if (my\_char == '\n')

{

num\_line++;

}

}

cout << "\nNumber of vowels : " << vowel << endl << "\nNumber of lines : " << num\_line;

\_getch();

return 0;

}

**Exercise 4**

Write a program that inputs the names of five cities from user in a string and stores/write them in a file city.txt

**Code :**

#include<iostream>

#include<conio.h>

#include<fstream>

#include<string>

using namespace std;

int main()

{

string city\_name;

cout << "Creating a File\n\n\n";

ofstream City;

City.open("City.txt");

cout << "Enter Five Cities Name : \n";

for (int i = 1; i <= 5; i++)

{

cin >> city\_name;

City << city\_name;

}

City.close();

cout << "File is created";

\_getch();

return 0;

}

**Exercise 5**

Write a program that reads the records stored in students.txt file and display them. (Save the records/data of 3 students yourself in the file manually)

**Code:**

#include<iostream>

#include<conio.h>

#include<fstream>

using namespace std;

int main()

{

char my\_char;

ifstream fin;

fin.open("Students.txt");

while (!fin.eof())

{

fin.get(my\_char);

cout << my\_char;

}

fout.close();

\_getch();

return 0;

}