Aror University of Art, Architecture, Design & Heritage Sukkur.

Department of Artificial Intelligence and Multimedia Gaming

Fundamentals of Programming (Fall-2023)

LAB No. 11

Prepared by: Abdul Haseeb Shaikh

Objective of Lab No. 11:
After performing lab 11, students will be able to:

- Declare and use pointers in C++
- Create and Open Files in C++
- Read a file in C++
- Write to a file in C++
- Append to a file in C++
- Use Filing in functions and with arrays

Pointers:

- A variable which holds the memory address of other variables of same data type. Pointers are symbolic representation of memory addresses.

Example:

```
int main(){
int a=5;
int* b=&a;
cout<<b<<endl;
return 0;
}
```

```
C:\Users\92306\Desktop\Aror
0x70fe14

--------------------------------
Process exited after 0.05116 seconds with return value 0
Press any key to continue . . .
```

Dereference Operator (*):
It gives the value stored at any memory address.

```
#include<iostream>
using namespace std;

int main(){
string a="Hello";
string* b=&a;
cout<<*b<<endl;
return 0;
}
```

```
C:\Users\92306\Desktop\Aror
Hello

--------------------------------
Process exited after 0.04887 seconds with return value 0
Press any key to continue . . .
```

File Handling:

- A mechanism to store the output of a program in a file, get the data in a program from a file. Files help store these data permanently on a storage device.
- fstream is a header file which gives access to file handling methods in C++.
- Three classes of fstream are:
  - offstream (Helps to write data to the file, also known as output stream)
  - ifstream(Helps to read data from the file, also called input stream)
  - fstream(Combination of ifstream and ofstream, provides capability of reading and writing a file)

File Operations:

1. open() – This is used to open/create a file.
2. read() – This is used to read the data from the file.
3. write() – This is used to write new data to file.
4. close() – This is used to close the file.

Modes for opening a file:

Mode	Description
iso::in	File opened in reading mode
iso::out	File opened in write mode
iso::app	File opened in append mode

Code for opening/creating a file:

```
int main(){
fstream File;
File.open("A.txt",ios::in);

if(!File){
    cout<<"Error";
}
else{
    cout<<"Found the file";
    File.close();
}
return 0;
}
```

Code for reading the file:

```
int main(){

ifstream ReadFile("C.txt",ios::in);
string text="";
while(getline(ReadFile,text)){
    cout<<text;
}

return 0;
}
```

Code for writing data to the file:

```
#include<iostream>
#include<fstream>
using namespace std;
int main(){

    ofstream MyFile("FileName.txt");
    MyFile<<"Hello Filing"<<endl<<"we are doing good";

    MyFile.close();

    return 0;
}
```

Code for appending data to the file:

Appending is same to writing, but we need to provide the file mode as ios::app

Lab Exercises:

1. **Create a String variable called season, which stores current season value, create another variable called ptr_to_season, which stores reference to the season variable, finally print ptr_to_season, and dereference ptr_to_season.**

2. **Define a function in C++ named create_file, this function takes a string parameter called file_name, and then creates a text file called file_name.**

3. **Define a function in C++ called read_file, which takes a string parameter called file_name, this function will read the contents of file_name and then display each line from the file_name.**

4. **Define a function in C++ called write_file, which takes a string parameter called file_name and another parameter which is an string array called file_arr, this function will append the contents of the file_arr into file_name file.**

5. **A. Create a file using create_file function in side the main function.**

   B. Create a string array called questions/answers in main function and pass this array to write_file function along with file_name, the array should contain any 5 questions and their answers in MCQS format.

6. **Use system("CLS") function to clear the screen and use switch case statement to ask whether the user is a common user or an admin user.**
   a. **If the user is a common user then take quiz from the user, you will use readfile function to read the contents of the file and take input for each question (the correct answers will be stored in another array called keys) and finally display the marks obtained by the user.**
   b. **If the user is an admin, ask the user whether wants to display the questions from the question bank, or wants to add some more questions in the question bank. For adding more questions into the question bank, you are going to use a loop which continuously takes string input from the user, opens a file and appends it inside the file, the loop will continuously take input until user gives 'n' as an input only.**