LAB #9
OS EXERCISE

Fortune Ogulewe
Austin hart
Adeniji wright
Emmanuel Odejobi

LAB 9 GROUP WORK

OS EXERCISE

Table of contents

1.0 Introduction

1.1 System Overview

This project involves portable file manipulation (pofm). File manipulation is the ability of the operating system to allow programmes to operate on files while also allowing the user to create, delete, rename, copy, move a file from one directory to another, add text to the end of a file, insert text in a specific location, remove all text from a file, and display the contents of a file.
File manipulation has a lot of advantages but it does have disadvantages as well. The advantages are as follows:
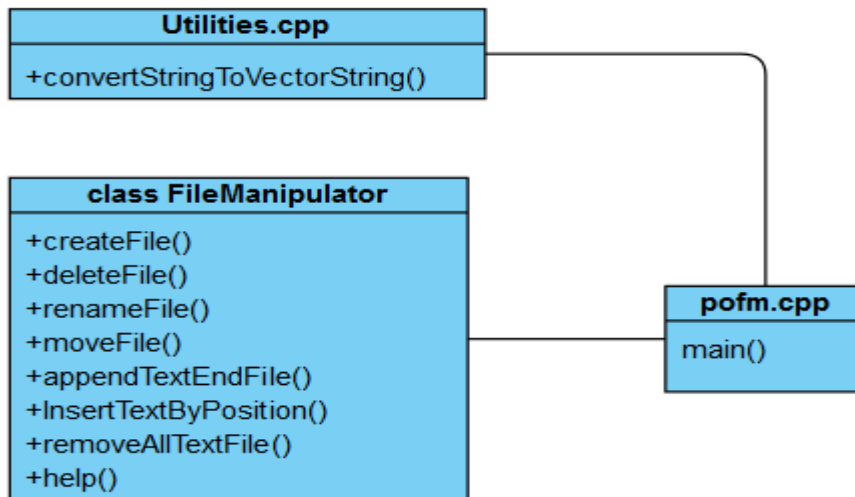
Advantages:
- Portability :-
  Since the program does not use any sort of system calls, it is not exclusively usable to just the Windows OS. The use of C++ language allows for usage in other platforms as well.
- Accessibility:-
  Opening of file for reading and writing via the internal stream buffer which supports input and output operations.
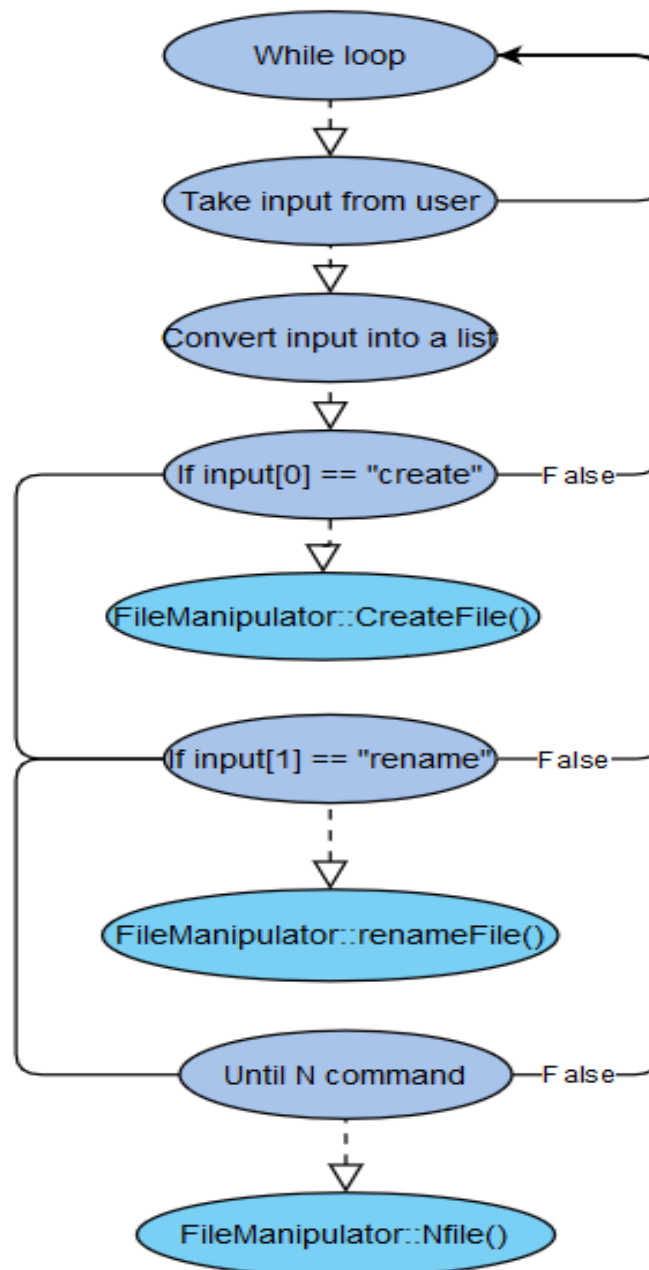
Disadvantages:
- Data Redundancy :-
  Where data is inserted more than once, and there is no method of validating whether there are duplicates in the system.

- Data Mapping and Access:-
  All related information may be grouped and stored in the system, no mapping between two files, that is two independant files are not linked. This is an inefficient and effective method to access data.

1.2 Design system and implementation


```
┌─────────────────────────────────────┐
│            Utilities.cpp             │
├─────────────────────────────────────┤
│ +convertStringToVectorString()       │
└─────────────────────────────────────┘

┌─────────────────────────────────────┐
│         class FileManipulator        │
├─────────────────────────────────────┤
│ +createFile()                        │
│ +deleteFile()                        │
│ +renameFile()                        │     ┌──────────────────┐
│ +moveFile()                          │     │     pofm.cpp      │
│ +appendTextEndFile()                 │     ├──────────────────┤
│ +InsertTextByPosition()              │     │ main()           │
│ +removeAllTextFile()                 │     └──────────────────┘
│ +help()                              │
└─────────────────────────────────────┘
```

The system is made in a very straightforward way. It consists of a class called

FileManipulator, which houses all the operations for modifying and manipulating files. There

is also the Utilities cpp, which has routines to make a variety of tasks simpler, but which is

unrelated to file manipulation. In light of this, the FileManiupulator class and the actual usage

of its functions are contained in the pofm.cpp file, which enables the user to manipulate files.

This method was chosen because it is straightforward and enables better code structure by

bundling or grouping together in the class functions that are linked to one another.

Additionally, it permits code reuse. It would be challenging to extract code that can be

utilised in other places if everything were implemented in main().

pofm.cpp

The input is taken as a string and then transformed into a list in the main application. This strategy was chosen because it makes it simpler for functions to interact with lists that have the command at index 0 and all the other arguments. The input is divided into a list, and the main function takes the element at index 0 and sequentially compares it with each if statement until one of the if statements is true, at which point it enters the loop and executes the appropriate function. Because it's straightforward, this implementation was picked.

2.0 User Manual

2.1 Create a file

Name:

createFile : create a new file

Function Synopsis:

createFile `nameofFile`
createFile `nameofFile1 nameNameFile2….nameOfFileN`

Description:

When this function is involved with any name, it will create a file in the current directory of the user. Also, it can be invoked with multiple files names and they will all be created in the same directory.

Screenshot:

Creating multiple files


```
create file1 file2 file3
Successfully created file1.pofm
Successfully created file2.pofm
Successfully created file3.pofm
```

The result

file1	2020-11-22 10:20 PM	POFM File	0 KB
file2	2020-11-22 10:20 PM	POFM File	0 KB
file3	2020-11-22 10:20 PM	POFM File	0 KB

Error handling

```
create file1
file1.pofm already exists. Please use another name.
```

2.2 Rename a File

Name:
renameFile: rename an existing file

Function Synopsis:
```
rename oldFileName
rename oldFileName1, oldFileName2, …,oldFileName(N)
```

Function Description:

The function accepts the oldFileName and checks if the file exists. If the file exists, it requests for a newFileName. Hence, enabling it to rename multiple files.

Screenshots
Renaming multiple files
```
File to be deleted cannot be located
rename file1 file2
rename a file.....
Enter the new File Name for file1.pofm
file4
```

Result

file2.pofm	2020-12-03 6:18 PM POFM File	0 KB
file3.pofm	2020-12-03 6:18 PM POFM File	0 KB
file4.pofm	2020-12-03 6:18 PM POFM File	0 KB
file5.pofm	2020-12-03 6:09 PM POFM File	0 KB

Error Handling
```
Enter the new File Name for file2.pofm
file5
Error in renaming files. Please close the file before renaming it.
```

2.3 Delete a File

Name:
deleteFile : Delete an existing file.

Function Synopsis:

```
delete fileName
delete fileName1, fileName2, …, fileName(N)
```

Function Description:

The function accepts the fileName to be deleted from the user, checks if the file exists and requests the user for double confirmation before deleting the file.

Screenshots
Deleting a file


```
delete file3
Are you sure you want to delete file3.pofm (y/n)y
Successfully deleted file3.pofm
```

Result

file4.pofm	2020-12-03 6:09 PM	POFM File		0 KB
file5.pofm	2020-12-03 6:09 PM	POFM File		0 KB

Error Handling


```
delete file4
File to be deleted cannot be located
```

2.4 Copy a File

Name:
copyFile : Copy a file from an existing file.

Function Synopsis:

```
copy sourceNameandDirectory, locationNameandDirectory
```
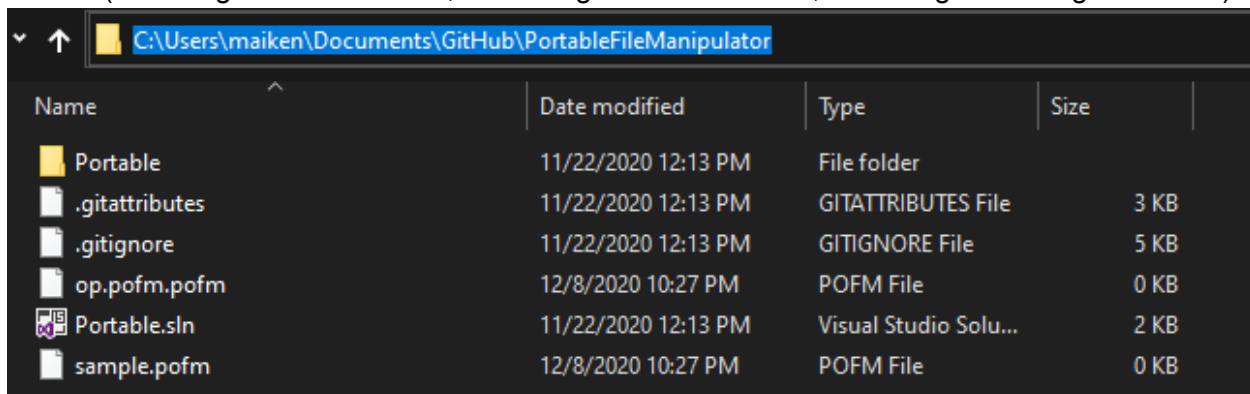
Function Description:
The function accepts two arguments: the fileName to be copied as well as the location where to put the copied file. Function checks if the file exists and gives appropriate warning if file does not exist.
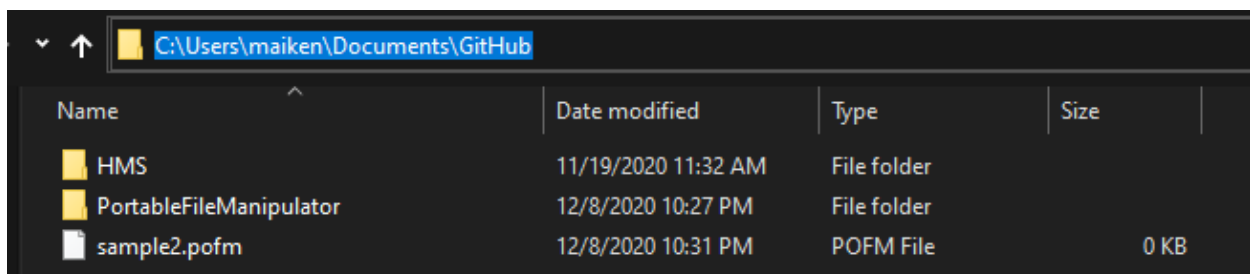
Screenshots

```
copy C:\Users\maiken\Documents\GitHub\PortableFileManipulator\sample.pofm
C:\Users\maiken\Documents\GitHub\sample2.pofm
```
(1st string is the command, 2nd string is the source file, 3rd string is the target location)

Name	Date modified	Type	Size
Portable	11/22/2020 12:13 PM	File folder	
.gitattributes	11/22/2020 12:13 PM	GITATTRIBUTES File	3 KB
.gitignore	11/22/2020 12:13 PM	GITIGNORE File	5 KB
op.pofm.pofm	12/8/2020 10:27 PM	POFM File	0 KB
Portable.sln	11/22/2020 12:13 PM	Visual Studio Solu...	2 KB
sample.pofm	12/8/2020 10:27 PM	POFM File	0 KB

C:\Users\maiken\Documents\GitHub\PortableFileManipulator

(source file: sample.pofm)

C:\Users\maiken\Documents\GitHub

Name	Date modified	Type	Size
HMS	11/19/2020 11:32 AM	File folder	
PortableFileManipulator	12/8/2020 10:27 PM	File folder	
sample2.pofm	12/8/2020 10:31 PM	POFM File	0 KB

(copied file: sample2.pofm)

Error Handling
```
copy
Missing two args
```
(missing source and target directory)
```
copy C:\Users\maiken\Documents\GitHub\PortableFileManipulator\sample.pofm
Missing one arg
```

(missing target directory)

2.5 Move a File

Name:
moveFile: Move a file from one directory to another.

Function Synopsis:
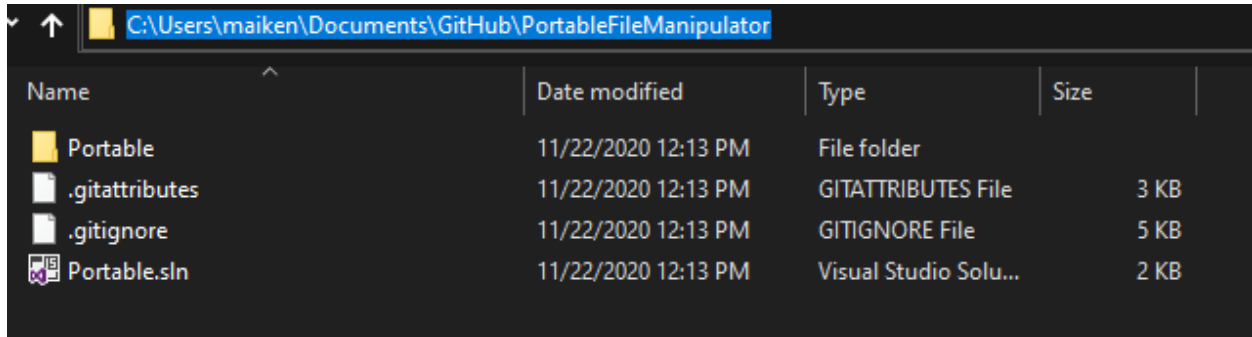`move sourceNameandDirectory, locationNameandDirectory`

Function Description:
The function accepts two arguments: the fileName to be moved as well as the location where to put the file. Function checks if the file exists and gives appropriate warning if file does not exist.
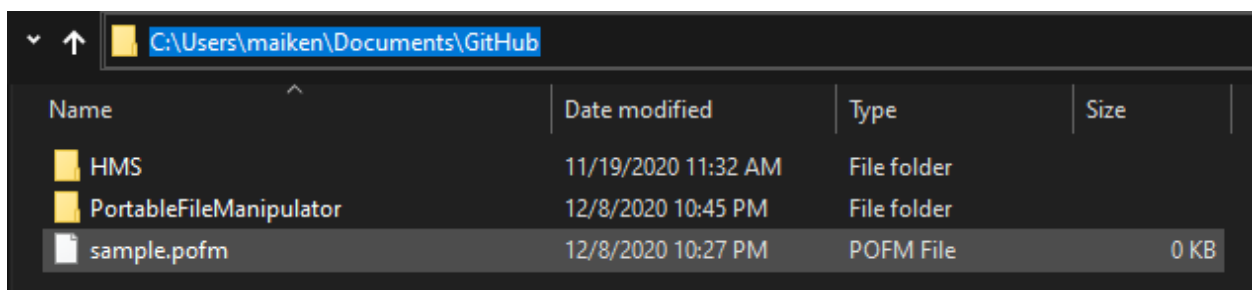
Screenshots:

(moved file: sample.pofm)

Error Handling:

```
move
Missing an args
```
(missing source directory and target directory)
```
move C:\Users\maiken\Documents\GitHub\PortableFileManipulator\sample.pofm
Missing one more arg
```
(missing target directory)

2.6 Append to end of the File

Name:

append : It takes text from the user and appends to the end of the file.
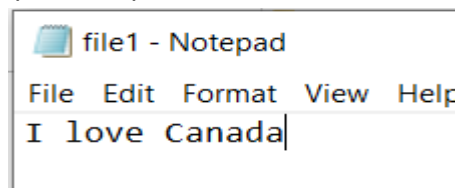
Function Synopsis:

```
append fileName 'text'
```

Function Description:

The function accepts the name of file as input and as well the text to be appended at the end of file. The text to be appended must between two single quotations marks otherwise the software will through an error message.
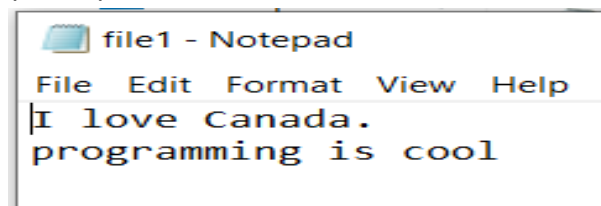
Screenshots:

(Before)The inside of file1

```
file1 - Notepad
File  Edit  Format  View  Help
I love Canada
```

Append to the file

```
append file1 'programming is cool'
```

(After)The inside of file1

```
file1 - Notepad
File  Edit  Format  View  Help
I love Canada.
programming is cool
```

Error Handling:

```
append file1
Missing one arg
```

2.7 Insert text in specified position

Name:

Insert : It takes text from the user and inserts based on the specified position.
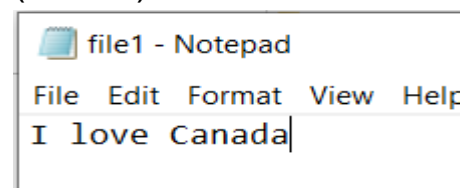
Function Synopsis:

```
insert fileName postion 'text'
```

Function Description:

The function accepts the name of file as input and specified position as well the text to insert.
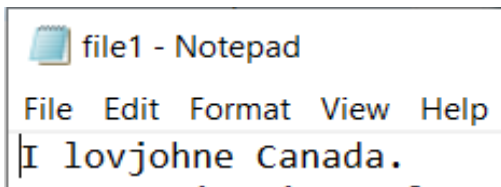
Screenshots:

(Before)The inside of file1

```
file1 - Notepad
File  Edit  Format  View  Help
I love Canada
```

Insert to the file

```
insert file1 5 'john'
```

(After)The inside of file1

file1 - Notepad

File Edit Format View Help

I lovjohne Canada.

Error Handling:

Missing three arguments: filename, position, text

```
insert
Missing three args
```

Missing two arguments: position and text

```
insert file1
Missing an position arg and text
```

Missing the position

```
insert file1 5
Missing one args
```

2.8 Remove text in the file

Name:

removedata : It removes all the data in the file.

Function Synopsis:

```
removedata fileName
removedata fileName1, fileName2, …, fileName(N)
```

Function Description:

The function accepts either a single file or multiple files and it removes all the data inside the file.

Screenshots:

(Before)The inside of file1

file1 - Notepad

File   Edit   Format   View   Help

I lovjohne Canada.
programming is cool

Removedata from file

removedata file1

(After)The inside of file1

file1 - Notepad

File   Edit   Format   View   Help

Error Handling:

removedata
Missing one arg

2.9 Help Function

Name:
removedata : It list out all the available functions

Function Synopsis:

```
Help
Help create
Help rename
Help delete
```

```
Help rename
Help removedata
Help append
Help insert
Help copy
```

Function Description:

The function lists out all the available commands, gives a brief summary of what is functionality of the command and how to use it with some examples.

Screenshots:

Help listing out all the available commands

Help with insert

Help with insert

Error Handling:

There is no error handling.