ECS793P - Introduction to Object Oriented Programming 19/20
Coursework Report
(Computer Store)

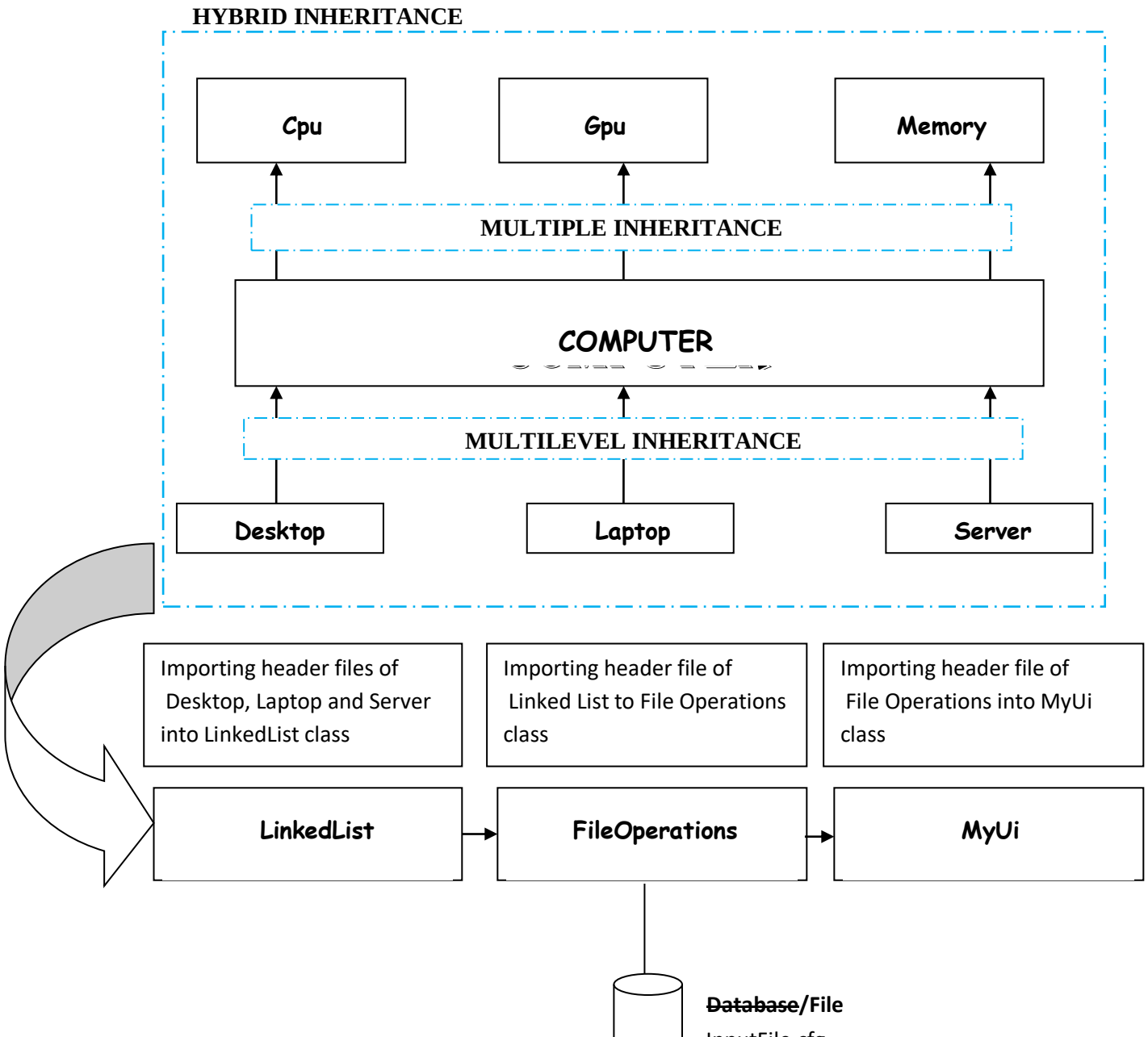
Sinnadurai Mahendran Pravin Durai
(190783415)

Introduction

For this coursework we use C++ to implement a computer store. The computer store has various types of computers such as Desktop, Laptops and Servers. Each having their own specification.
For storing the data files were used instead of database.
We have used Linked List for storing and retrieving the information from the file. We have customized the data part of the Linked list so that it can store objects of Desktop, Laptop, Server.

Class Hierarchy

MULTIPLE inheritance - We have a Class called "Computer" inherited from the class called Cpu, Gpu and Memory
MULTILEVEL inheritance – The Classes "Desktop, Laptop and Server" are inherited from "Computer" and "Computer" is inherited from "Cpu,Gpu,Memory"
The combination of these two makes the Hybrid Inheritance



Function Explanation

The below table shows some of the important function that i have created for each class, where the actual processing takes place.
The detailed information about all the function is available in the Excel file “Test Case for Computer Store”under the tab “FunctionExplanation”. You can find this file as a part of the project zip file.

#Sno	Class	Function	Explanation
1	MyUi	create GUI1	Creates the First screen that shows the List of Computers (Desktop, Laptop, Server)
2		createGUI2	It displays the list of operation that can be performed by the user for a given computer **NOTE:** Functionality are different for different type of Computers
3		generateId	This method is used to automatically generate the System ID, And it's always Unique
4	FileOperations	listToFile	This is one of the important function for the effecrtive functioning of the Application. It stores the data from the List to the File before it exist the application
5		fileToList	This method reads the contents of the file and stores them in the List before any other function gets executed. Post this all the computations are done only in List.
6		getMaxSysId	This Method is used to find the Maximum value of the system ID that is present in the system.
7		addNewNode	This method is used to add new Node to the existing Linked List
8		displayAllNodes	Although the method name says 'displayAllNodes', This method display only the nodes that are belonging to a particular system
9		displayCDescription	Used to display the detailed bill for the product along with their Specifications
10	Desktop	getAllInputs	This function get the Input from the user and stores them in the object
11		displayDesktopInputs	Display the inputs received from a particular object to the screen
12		desktopCostDescription	This fnunction display the destop cost and it's specification to the ouput screen
13		calculateDesktopCost	It calculate the cost for the Desktop, But in actual it make a call to the function that is available in super class to calculate the cost.

14	Laptop	getAllInputs	This function get the Input from the user a nd stores them in the object
15		displayLaptopInputs	Display the inputs received from a particular object to the screen
16		LaptopCostDescription	This fnunction display the destop cost and it's specification to the ouput screen
17		calculateLaptopCost	It calculate the cost for the Laptop, But in actual it make a call to the function that is available in super class to calculate the cost.
18	Server	getAllInputs	This function get the Input from the user and stores them in the object
19		displayServerInputs	Display the inputs received from a particular object to the screen
20		ServerCostDescription	This fnunction display the destop cost and it's specification to the ouput screen
21		calculateServerCost	It calculate the cost for the Server, But in actual it make a call to the function that is available in super class to calculate the cost.
22	Computer	displayCompInputs	This function displays all the inputs that below to Computer class (Eg : CPU, GPU and Memory related info)
23		displayCompCostInputs	This function Display the Computer Specification along with its cost
24		costCalculator	This is were the actual cost is calculated no matter what type of Computer it is. (Desktop, Laptop, Server)
25		validateStringInput	This function validates whether the given Input is a integer or a String

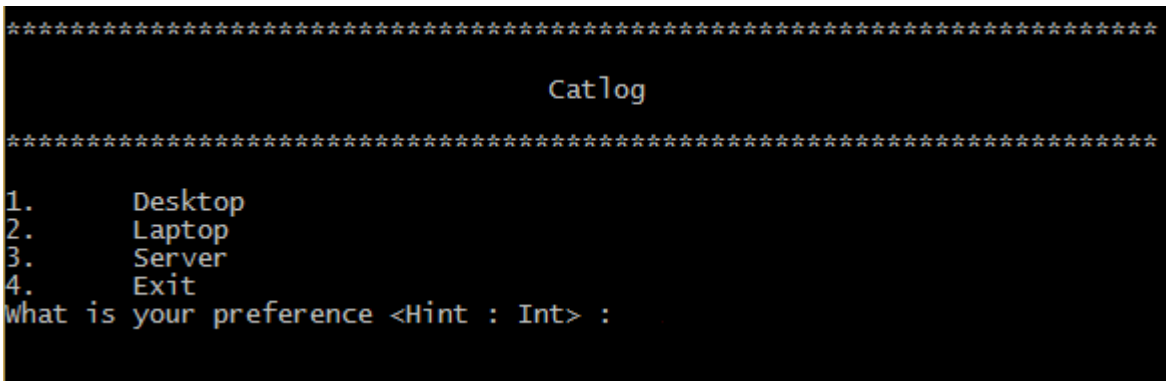
Test

The Application Under Test was tested with various inputs to see if there is any functionality that is not working as expected. The attached Excel sheet contains both positive and negative test cases (The negative test cases are highlighted in Red colour) that are covered during the testing process. This Excel file is also available in the Zip file.

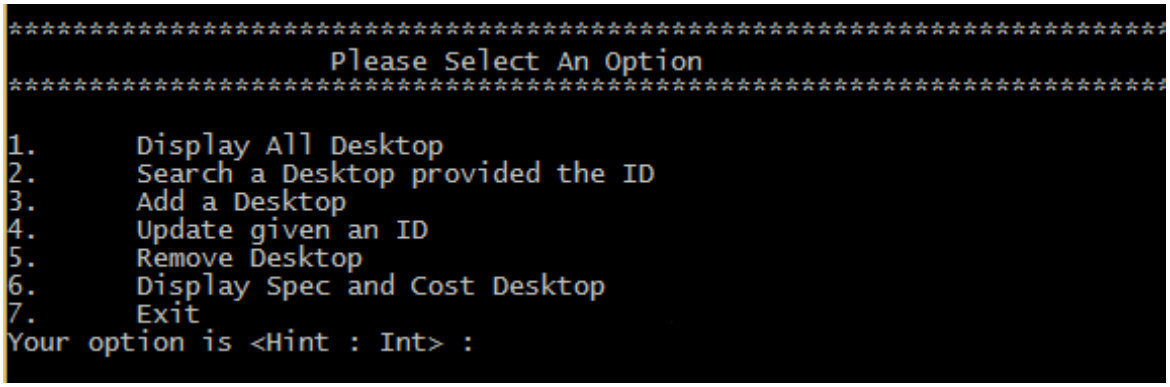
Test Case for Computer Store.xlsx

Sample Outputs from the Application

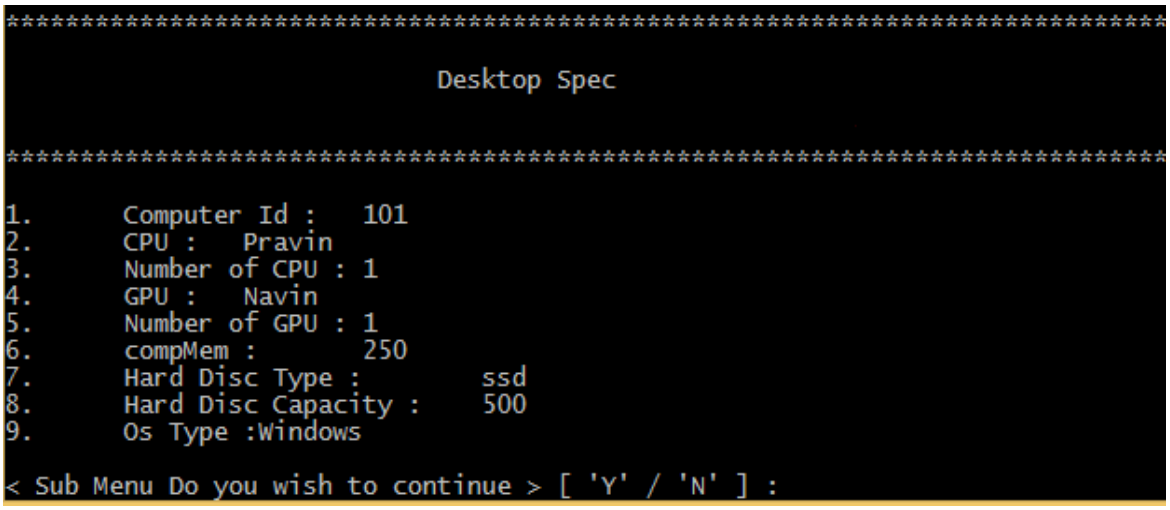
GUI 1



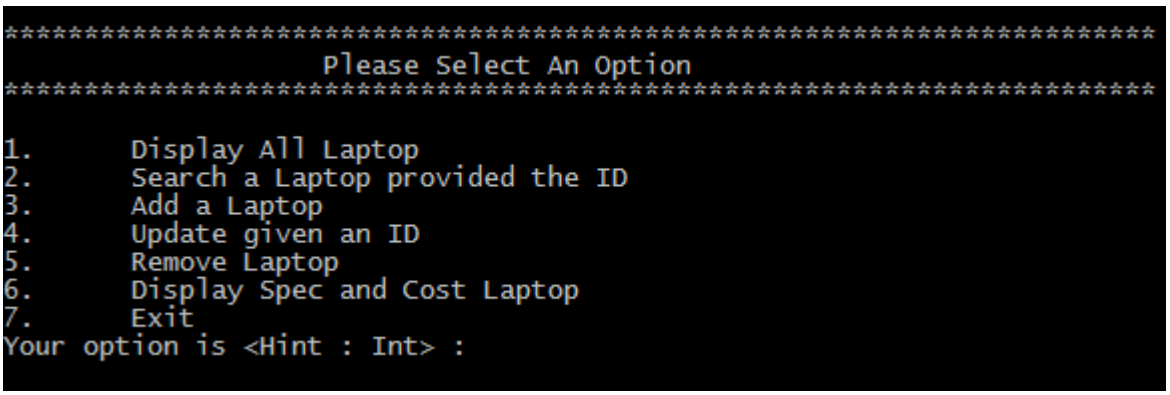
GUI 2
Test Case 2.1



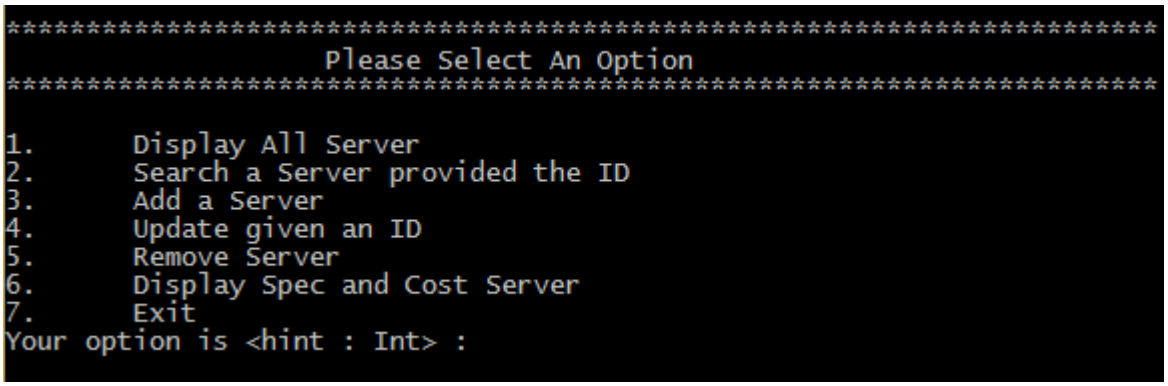
Test Case 2.1.2



Test Case 2.2



Test Case 2.3



6. Limitations

Constraint on user input

The function interface is designed as simple as possible there are certain scenario where the user will be prompted to choose between **'Y' or 'N'**, In this case if the user inputs a character then the application works as expected. But if the users enters a string or an Integer the system prints the same line asking for the inputs based on the length of the string given as input by the user. But the system will not crash.

Programming Environment: Windows
Compiler used: realgcc.exe (Rev1, Built by MSYS2 project) 7.2.0
Way to compile the code in console: >> g++ Main.cpp MyUi.cpp LinkedList.cpp FileOperations.cpp Desktop.cpp Laptop.cpp Server.cpp Computer.cpp Cpu.cpp Gpu.cpp Memory.cpp -o Main
To run the program: >>Main

Note: *It's good if you can try in window, Coz, i get some compilation errors when i tried compiling the code in Linux environment .Also certain methods used for clearing the console might be restricted to windows functionality.*
The code works perfectly fine in windows.