Individual Assessment Coversheet

To be attached to the front of the assessment.

Campus: Tygervalley

Faculty: Technology

Module Code: ITPPAB44

Group: 4

Lecturer's Name: Jealous Kamwaya

Student Full Name: Maile Omphile Chokoe

Student Number: EDUV4825799

Indicate	Yes	No
Plagiarism report attached		

Declaration:

I declare that this assessment is my own original work except for source material explicitly acknowledged. I also declare that this assessment or any other of my original work related to it has not been previously, or is not being simultaneously, submitted for this or any other course. I am aware of the AI policy and acknowledge that I have not used any AI technology to generate or manipulate data, other than as permitted by the assessment instructions. I also declare that I am aware of the Institution's policy and regulations on honesty in academic work as set out in the Conditions of Enrolment, and of the disciplinary guidelines applicable to breaches of such policy and regulations.

Signature	**Date**

Lecturer's Comments:

Marks Awarded:	**%**

Signature	**Date**

Question 1

```
#include <iostream>
#include <string>
#include <fstream>

using namespace std;

//discounts
const double DISCOUNT_RATE = 0.10; // 10% discount
const double DISCOUNT_THRESHOLD = 100.00; // Discount applies if bill is over R100

//Menu Prices
const double coffee_p = 15.00;
const double sandwich_p = 30.00;
const double salad_p = 25.00;
const double juice_p = 10.00;
const double muffin_p = 20.00;
const double pizza_p = 35.00;
const double soup_p = 18.00;
const double burger_p = 40.00;

//glb variables
bool itemvalid = false;
int iItemNum = 0, sum = 0, picked = 0;


void static drawMenu() {
    //Second Bullet Point
```

```
    cout << "Menu Prices:" << endl;
    //By using raw string literals aka R"()", we save alot of time and effort
    cout << R"(
1. Coffee - R15.00
2. Sandwich - R30.00
3. Salad - R25.00
4. Juice - R10.00
5. Muffin - R20.00
6. Pizza - R35.00
7. Soup - R18.00
8. Burger - R40.00
)";
}

void static selection() {

    //Third Bullet Point
    cout << "Select how many items you would like to select (1-8): ";

    while (itemvalid == false) {
        cin >> iItemNum;
        if ((iItemNum < 1) || (iItemNum > 8)) {
            cout << "Please select a valid input: ";
        }
        else {
            itemvalid = true;
        }
```

```
    }

    //the loop of the selection
    //Item num = Number of items;   Sum is self evident; picked = which item is picked

    //fourth bullet point
    for (int i = 0; i < iItemNum; i++) {
        cout << "Select Item (" + to_string(i+1) + "): ";
        cin >> picked;

        if (picked == 1) {
            sum = sum + coffee_p;
        }
        else if (picked == 2) {
            sum = sum + sandwich_p;
        }
        else if (picked == 3) {
            sum = sum + salad_p;
        }
        else if (picked == 4) {
            sum = sum + juice_p;
        }
        else if (picked == 5) {
            sum = sum + muffin_p;
        }
        else if (picked == 6) {
            sum = sum + pizza_p;
        }
```

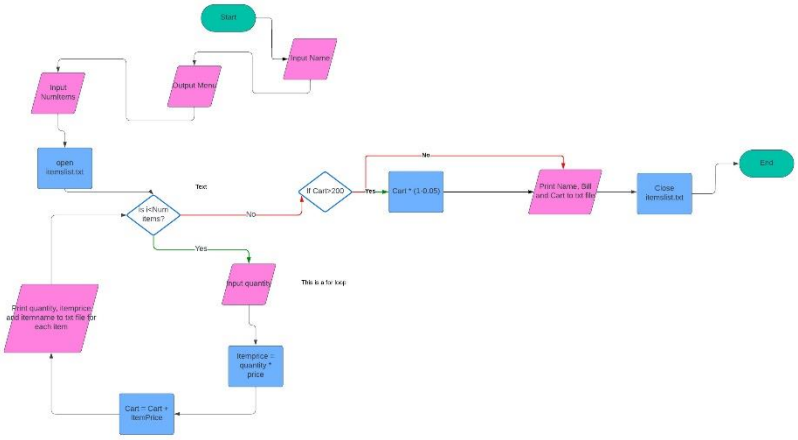
```
        else if (picked == 7) {
            sum = sum + soup_p;
        }
        else if (picked == 8) {
            sum = sum + burger_p;
        }
    }

        //fifth and sixth
        cout << "\nTotal Bill: R" + to_string(sum) << endl;

        if (sum > DISCOUNT_THRESHOLD) {
            cout << "Discount will be applied" << endl;
            sum = sum * (1 - DISCOUNT_RATE);
        }
        else {
            cout << "Discount will not applied" << endl;
        }

        cout << "\nFinal Bill: R" + to_string(sum) << endl;
}

//the main function. We split up the code to make this nicer to read

int main() {
    //First bullet point
    string name, surname;

    cout << "Please enter your name: ";
```

```
cin >> name;

cout << "Please enter your surname: ";
cin >> surname;

drawMenu();

selection();

//open our input file
ofstream OutFile;
OutFile.open("cafeteriabill.txt");

//writing to the file
if (OutFile.is_open()) {
    OutFile << name + " " + surname + " ";
    OutFile << "The final bill was R" + to_string(sum) << endl;
    cout << "Your bill has been written to cafeteriabill.txt";

    OutFile.close();
}

else {
    perror("File opening failed"); // (IBM I 7.3, n.d.)
    cout << "Therefore, there were issues opening a file.\n " << endl;
}
```

```
    return 0;
}
```

Question 2

2.1.1



2.1.2)

```
#include <iostream>
#include <string>
#include <fstream>

using namespace std;

int main() {
        string name = "", itemName = "";
        int numItems = 0, quantity = 0, itemPrice = 0, cart = 0;

        cout << "Hello, please input your name and surname: ";
        getline(cin >> ws, name);

        //drawing the menu
```

```
cout << "\nList of available items: " << endl;

cout << "Red Apples                    R5" << endl;
cout << "Chicken Breasts               R54" << endl;
cout << "Hot Chocolate Powder          R92" << endl;
cout << "Still Water                 R25" << endl;
cout << "Lays Chips                  R22" << endl;
cout << "Red Grapes                    R29" << endl;
cout << "Beef Rump Steak R118" << endl;

//done drawing the menu

cout << "\nHow many of each item do you want?: ";
cin >> numItems;
cout << endl;
//open the text file
ofstream TFile; //open file (Cplusplus.com, D. nd)
TFile.open("itemslist.txt");

if (TFile.is_open()) {

        TFile << name << endl;
        for (int i = 0; i < numItems; i++) {
                cout << "Select an item: ";
                getline(cin >> ws, itemName);
                cout << endl;

                cout << "How much of this item do you want?: ";
                cin >> quantity;
```

```
                cout << endl;


                if (itemName == "Red Apples" || itemName == "red apples") {
                        itemPrice = 5 * quantity;
                        cart = cart + itemPrice;
                        TFile << itemName + "(x" + to_string(quantity) + ")
        " + "R" + to_string(itemPrice) << endl; //we add to the text file line by line


                }
                else if (itemName == "Chicken Breasts" || itemName == "chicken
breasts") {
                        itemPrice = 54 * quantity;
                        cart = cart + itemPrice;
                        TFile << itemName + "(x" + to_string(quantity) + ")
        " + "R" + to_string(itemPrice) << endl;
                }
                else if (itemName == "Hot Chocolate Powder" || itemName ==
"hot chocolate powder") {
                        itemPrice = 92 * quantity;
                        cart = cart + itemPrice;
                        TFile << itemName + "(x" + to_string(quantity) + ")
        " + "R" + to_string(itemPrice) << endl;
                }
                else if (itemName == "Still Water" || itemName == "still water") {
                        itemPrice = 25 * quantity;
                        cart = cart + itemPrice;
                        TFile << itemName + "(x" + to_string(quantity) + ")
        " + "R" + to_string(itemPrice) << endl;
                }
                else if (itemName == "Lays Chips" || itemName == "lays chips") {
                        itemPrice = 22 * quantity;
```

```
                cart = cart + itemPrice;
                TFile << itemName + "(x" + to_string(quantity) + ")
        " + "R" + to_string(itemPrice) << endl;
            }
            else if (itemName == "Red Grapes" || itemName == "red
grapes") {
                itemPrice = 29 * quantity;
                cart = cart + itemPrice;
                TFile << itemName + "(x" + to_string(quantity) + ")
        " + "R" + to_string(itemPrice) << endl;
            }
            else if (itemName == "Beef Rump Steak" || itemName == "beef
rump steak") {
                itemPrice = 118 * quantity;
                cart = cart + itemPrice;
                TFile << itemName + "(x" + to_string(quantity) + ")
        " + "R" + to_string(itemPrice) << endl;
            }
            else {
                cout << "That is not an actual item, please input again"
<< endl;
                i = i - 1;
            } // if else chain
        } // for loop
    } //if open?
    else {
        perror("File opening failed");
    }

    if (cart > 200) {
        cout << "Discount available!" << endl;
```

```
            cart = cart * (1 - 0.05);

        }



        cout << "\nR" + to_string(cart) ;
        TFile << "------------------------------\n" << endl;
        TFile << "Total Bill: " + to_string(cart) << endl;

        TFile.close(); // close file




        return 0;
}
```

Question 3

```
#include <iostream>
#include <string>
int arrScores[5] = { 0, 0, 0, 0, 0 }; // a)

//function average scores

float average() {
        float sum = 0;

        for (int i = 0; i <= 4; i++) {
```

```
                sum = sum + arrScores[i];
        }

        return sum/5;
}

//function find and return largest number
int highest() {
        int largest = arrScores[0];

        for (int i = 0; i <= 4; i++) {
                if (arrScores[i] > largest) {
                        largest = arrScores[i];
                }
        }
        return largest;
}

//function find and return lowest number
int lowest() {
        int smallest = arrScores[0];
        for (int i = 0; i <= 4; i++) {
                if (arrScores[i] < smallest) {
                        smallest = arrScores[i];
                }
        }
        return smallest;
}
```

```
int main() {

	std::cout << "Enter the scores of 5 students:\n" << std::endl;

	for (int i = 0; i <= 4; i++) { // entering the scores
		std::cout << "Enter the score of student " + std::to_string(i+1) + ": ";
		std::cin >> arrScores[i];



	}
	std::cout << std::endl;

	for (int i = 0; i <= 4; i++) { // entering the scores
		std::cout << "Student " + std::to_string(i + 1) + ": " +
std::to_string(arrScores[i]) << std::endl;
	}
	std::cout << std::endl;




	std::cout << "Average Score: " + std::to_string(average()) << std::endl; //
average
	std::cout << "Highest Score: " + std::to_string(highest()) << std::endl; // largest
	std::cout << "Lowest Score: "  + std::to_string(lowest()) << std::endl; // lowest

	return 0;
}
```

Question 4

```
#include <iostream>
```

```
#include <string>

using namespace std;

void createAccount(string &name,int &accountNumber, double &balance) {
//Question 4.1.1
	cout << "Hello user, can you please type your full name: ";
	getline(cin >> ws, name);
	cout << endl;

	cout << "Now you must type a 10 digit account number: ";
	cin >> accountNumber;
	cout << endl;

	cout << "Now type your initial deposit please (Make sure it is greater than 0):
";
	cin >> balance;

	while (balance <= 0) {
		cout << "Now type your initial deposit please: ";
		cin >> balance;
	}
	cout << endl;

	cout << "Accout created successfully\n" << endl;

}

double depositMoney(double &balance) { //Question 4.2.1
	double userDeposit = 0;
```

```
	while (userDeposit <= 0) {
		cout << "How much money would you like to input into your account? (Input must be greater than 0): ";
		cin >> userDeposit;
		cout << endl;
	}
	balance = balance + userDeposit;
	return balance;
}

double withdrawMoney(double &balance) { //Questions 4.3.1 //(Udacity, 2024)
	double userWithdraw = 0;

	while (userWithdraw <= 0) {
		cout << "How much money would you like to withdraw from your account? (Must be greater than 0): ";
		cin >> userWithdraw;
		cout << endl;

	}
	balance = balance - userWithdraw;
	return balance;
}

string checkBalance(double& balance) { //Question 4.4.1
	return "Account Balance: " + to_string(balance);
}
```

```
string displayAccountDetails(const string& name, const int& accountNumber, const double& balance) {
	return "---Account Details--- \nAccount Holder: " + name + "\nAccount Number: " + to_string(accountNumber) + "\nCurrent Balance: " + to_string(balance);
}

void drawMenu() {
	cout << "\n--- Bank Account Management system --- \n" << endl;
	cout << "1. Create Account \n2.Deposit Money \n3.Withdraw Money \n4.Check Balance \n5.Display Account Details \n6.Exit" << endl;
}

int main() {
	int choice = 0;

	string name = ""; (W3 Schools, 2024)
	int accountNumber = 0;
	double balance = 0;

	while (choice != 6) {
		drawMenu();
		cout << "Enter your choice (1-6): ";
		cin >> choice;

		if (choice == 1) {
			createAccount(name, accountNumber, balance);
		}
		else if (choice == 2) {
			cout << "Current Balance: R" + to_string(depositMoney(balance)) << endl;
```

```
            }
            else if (choice == 3) {
                cout << "Current Balance: R" +
to_string(withdrawMoney(balance)) << endl;;
            }
            else if (choice == 4) {
                cout << "Current Balance: R" + checkBalance(balance) << endl;

            }
            else if (choice == 5) {
                cout << displayAccountDetails(name, accountNumber, balance)
<< endl;

            }

        }

        cout << "\nExiting this system, goodbye kind individual";
        return 0;
}
```

Extra

In the event you want a different way to access my code or are having issues with opening my files. Use the link to the GitHub Repository here: https://github.com/DaChoco/Block_4Project

Bibliography

IBM i 7.3. (D. nd.). https://www.ibm.com/docs/en/i/7.3?topic=functions-perror-print-error-message (Accessed 30 October)

Cplusplus (D. nd), Cplusplus.com, Input/Output with files, Available at: https://cplusplus.com/doc/tutorial/files/ (Accessed 30 October)

W3 Schools (2024) *W3schools.com*, *W3Schools Online Web Tutorials*. Available at: https://www.w3schools.com/cpp/cpp_pointers.asp (Accessed: 18 October 2024).

Udacity Team (2024) Udacity.com, C++ Dereferencing explained. Available at https://www.udacity.com/blog/2021/07/cpp-dereferencing-explained.html (Accessed 30 October 2024)