

```
// Shahad Hussain AL Dhaif 2230001147.
```

```
// Yara Ali Alshehri 2230006494.
```

```
// Fatimah Majid Almar 2230004016.
```

```
#include <iostream>
```

```
#include <string> //Yara Alshehri
```

```
#include <fstream> // Shahad Hussain AL Dhaif
```

```
using namespace std;
```

```
struct MenuItem {
```

```
    string name;
```

```
    float price;
```

```
};
```

```
struct OrderItem {
```

```
    MenuItem item;
```

```
    int quantity;
```

```
};
```

```
// Function prototypes
```

```
// Shahad AL Dhaif
```

```
void displayMenu(MenuItem menu[], int size, OrderItem order[], int& orderSize, float& total, int& loyaltyPoints, string& userName, string& studentID);
```

```
void choosePickupTime();
```

```
// Fatimah Majid Almar
```

```
void addItem(OrderItem order[], int& orderSize, MenuItem foodMenu[], int foodSize,
MenuItem drinksMenu[], int drinksSize, MenuItem dessertsMenu[], int dessertsSize,
float& total, int& loyaltyPoints, string& userName, string& studentID);
```

```
void removeItem(OrderItem order[], int& orderSize);
```

```
// Shahad AL Dhaif
```

```
void displayOrder(OrderItem order[], int orderSize);
```

```
// Yara Ali Alshehri
```

```
float calculateTotal(OrderItem order[], int size, int& loyaltyPoints);
```

```
float applyDiscount(float total, int points);
```

```
// Define menu items
```

```
// Shahad AL Dhaif
```

```
MenuItem foodMenu[] = {"French Fries", 7.0}, {"Falafel", 8.0}, {"Pizza", 13.0}, {"Salad",
7.0}, {"Grape Leaves", 8.0}};
```

```
MenuItem drinksMenu[] = {"Orange Juice", 2.0}, {"Lemon Juice", 2.0}, {"Apple Juice", 2.0},
{"Cherry Juice", 2.0}, {"Milk", 2.0}, {"Coffee", 10.0}, {"Tea", 7.5}};
```

```
MenuItem dessertsMenu[] = {"Ice Cream", 6.0}, {"Croissant", 11.0}, {"Pancake", 10.0},
{"Waffle", 10.0}, {"Cupcake", 10.0}};
```

```
// Shahad AL Dhaif
```

```
int main() {
```

```
    ofstream outfile;// Shahad Hussain AL Dhaif
```

```
    string userName;
```

```
    string studentID;
```

```
    float total = 0.0;
```

```

int loyaltyPoints = 0;

const int MAX_ORDER_SIZE = 100; // Maximum number of items in the order

OrderItem order[MAX_ORDER_SIZE]; // Array to store the order

int orderSize = 0; // Current size of the order


int choice;


    cout << "  [-_-] [+_-] Welcome to the university cafeteria ordering system! *  " <<
endl;

    cout << "\n";

    cout << "Please enter your Name: ";

    cin.ignore();

    getline(cin, userName); // Get user name

    cout << "Please enter your student ID: ";

    getline(cin, studentID); // Get student ID


do {

    cout << "\nOptions Menu:\n";

    cout << "1. Choose Menu\n";

    cout << "2. Add Item\n";

    cout << "3. Finish Order and Pay\n";

    cout << "4. Remove Item\n";

    cout << "5. Exit\n";

    cout << "Enter your choice: ";

    cin >> choice;


    switch(choice) {

        case 1:

```

```

cout << "Choose the menu you want to browse:\n";

cout << "1. Food Menu\n";

cout << "2. Drinks Menu\n";

cout << "3. Desserts Menu\n";

cout << "Enter menu choice: ";

cin >> choice;


switch(choice) {

    case 1:

        displayMenu(foodMenu, sizeof(foodMenu)/sizeof(foodMenu[0]), order,
orderSize, total, loyaltyPoints, userName, studentID);

        break;

    case 2:

        displayMenu(drinksMenu, sizeof(drinksMenu)/sizeof(drinksMenu[0]), order,
orderSize, total, loyaltyPoints, userName, studentID);

        break;

    case 3:

        displayMenu(dessertsMenu, sizeof(dessertsMenu)/sizeof(dessertsMenu[0]),
order, orderSize, total, loyaltyPoints, userName, studentID);

        break;

    default:

        cout << "Invalid choice!\n";

}

break;

case 2:// Fatimah Majid Almar. Add Item

    addItem(order, orderSize, foodMenu, sizeof(foodMenu)/sizeof(foodMenu[0]),
drinksMenu, sizeof(drinksMenu)/sizeof(drinksMenu[0]), dessertsMenu,
sizeof(dessertsMenu)/sizeof(dessertsMenu[0]), total, loyaltyPoints, userName,
studentID);

    break;

```

```

case 3: // Shahad Hussain AL Dhaif [I just added the file]

    // Yara Alshehri [Finish Order and Pay ]

    outfile.open("BILL(2).txt",ios::out);

    if(outfile.is_open()) {

        cout<<"\n - - - - -" << endl;

        cout<<"\n";

        cout<<"File is opened successfully !\n";

        total = calculateTotal(order, orderSize, loyaltyPoints); // Calculate total amount
and points : Yara Alshehri

        outfile << "   [-_-] [+_-] Welcome to the university cafeteria ordering system! *
" << endl;

        outfile << "\n";

        outfile << userName << " " << studentID << "! Your total bill is: " << total << "
SAR\n";

        outfile << "\n";

        outfile << "You earned " << loyaltyPoints << " points.\n";

        outfile << "\n";

        total = applyDiscount(total, loyaltyPoints); // Apply discount if applicable :Yara
Alshehri

        choosePickupTime();

        outfile << "Thank you for your order [*-_-]!\n";

        outfile << "\n";

        outfile << "Please pay " << total << " SAR.\n";

        outfile.close();


        // Reset order for the next customer

        orderSize = 0;

        loyaltyPoints = 0;

```

```

        } else {
            cout << "Failed to open the file!\n";
        }
        break;
    case 4:// Fatimah Majid Almar . Remove Item
        removeItem(order, orderSize);
        break;
    case 5:
        cout << "Exiting the program. Goodbye! [-`_`-] \n";
        break;
    default:
        cout << "Invalid choice. Please try again." << endl;
        break;
    }
} while(choice != 5);

return 0;
}

// Shahad AL Dhaif

void displayMenu(MenuItem menu[], int size, OrderItem order[], int& orderSize, float&
total, int& loyaltyPoints, string& userName, string& studentID) {
    cout << "Menu:\n";
    for(int i = 0; i < size; ++i) {
        cout << i+1 << ". " << menu[i].name << " - " << menu[i].price << " SAR\n";
    }

    int choice;

```

```
    cout << "Enter the number of the item you want to order (or 0 to go back to the main menu): ";
```

```
    cin >> choice;
```

```
    if(choice >= 1 && choice <= size) {
```

```
        cout << "Enter the quantity for " << menu[choice - 1].name << ": ";
```

```
        int quantity;
```

```
        cin >> quantity;
```

```
        // Add the item to the order :Shahad Al Dhaif
```

```
        order[orderSize].item = menu[choice - 1];
```

```
        order[orderSize].quantity = quantity;
```

```
        orderSize++;
```

```
        // Display the menu again :Shahad Al Dhaif
```

```
        displayMenu(menu, size, order, orderSize, total, loyaltyPoints, userName, studentID);
```

```
    } else if(choice == 0) {
```

```
        // Go back to the main menu :Shahad Al Dhaif
```

```
        return;
```

```
    } else {
```

```
        // Invalid choice :Shahad Al Dhaif
```

```
        cout << "Invalid choice!\n";
```

```
        displayMenu(menu, size, order, orderSize, total, loyaltyPoints, userName, studentID); // Redisplay the menu
```

```
        return;
```

```
    }
```

```
}
```

```
// Shahad AL Dhaif
```

```
void choosePickupTime() {
```

```
    cout << "Pickup Time Options:\n";
```

```
    cout << "1. Pickup now\n";
```

```
    cout << "2. Pickup in 30 minutes\n";
```

```
    cout << "3. Pickup in 1 hour\n";
```

```
    cout << "Enter your choice: ";
```

```
    int choice;
```

```
    cin >> choice;
```

```
    switch(choice) {
```

```
        case 1:
```

```
            cout << "Your order will be ready for pickup shortly.\n";
```

```
            break;
```

```
        case 2:
```

```
            cout << "Your order will be ready for pickup in 30 minutes.\n";
```

```
            break;
```

```
        case 3:
```

```
            cout << "Your order will be ready for pickup in 1 hour.\n";
```

```
            break;
```

```
        default:
```

```
            cout << "Invalid choice.\n";
```

```
            break;
```

```
    }
```

```
}
```

```
// Yara Ali Alshehri
```

```
float calculateTotal(OrderItem order[], int size, int& loyaltyPoints) {
```

```
    float total = 0.0;
```



```

for(int i = 0; i < size; ++i) {
    total += order[i].item.price * order[i].quantity;
}

// Calculate loyalty points based on the total amount spent :Yara Alshehri
if (total >= 30.0) {
    loyaltyPoints = 15;
} else if (total >= 20.0) {
    loyaltyPoints = 10;
} else if (total >= 10.0) {
    loyaltyPoints = 5;
} else if (total >= 5.0) {
    loyaltyPoints = 3;
} else {
    loyaltyPoints = 0;
}

return total;
}

```

```

// Yara Ali Alshehri
float applyDiscount(float total, int points) {
    if(points == 15) {
        cout << "You get a 10 SAR discount! Discount: 10 SAR\n";
        return total - 10.0;
    } else if(points == 10) {
        cout << "You get a 5 SAR discount! Discount: 5 SAR\n";
        return total - 5.0;
    } else if(points == 5) {

```

```

        cout << "You get a 3 SAR discount! Discount: 3 SAR\n";
        return total - 3.0;
    } else {
        cout << "You don't get any discount! Discount: 0 SAR\n";
        return total;
    }
}

```

// Fatimah Majid Almar

```

void addItem(OrderItem order[], int& orderSize, MenuItem foodMenu[], int foodSize,
MenuItem drinksMenu[], int drinksSize, MenuItem dessertsMenu[], int dessertsSize,
float& total, int& loyaltyPoints, string& userName, string& studentID) {

```

```

    int menuChoice;

```

```

    cout << "Choose the menu you want to add from:\n";

```

```

    cout << "1. Food Menu\n";

```

```

    cout << "2. Drinks Menu\n";

```

```

    cout << "3. Desserts Menu\n";

```

```

    cout << "Enter menu choice: ";

```

```

    cin >> menuChoice;

```

```

    switch(menuChoice) {

```

```

        case 1:

```

```

            displayMenu(foodMenu, foodSize, order, orderSize, total, loyaltyPoints,
userName, studentID);

```

```

            break;

```

```

        case 2:

```

```

            displayMenu(drinksMenu, drinksSize, order, orderSize, total, loyaltyPoints,
userName, studentID);

```

```

            break;

```

```

        case 3:
            displayMenu(dessertsMenu, dessertsSize, order, orderSize, total, loyaltyPoints,
                userName, studentID);

            break;

        default:
            cout << "Invalid choice!\n";

    }

    displayOrder(order, orderSize); // Display the updated order after adding an item
:Shahad Al Dhaif
}

```

```

// Fatimah Majid Almar

```

```

void removeItem(OrderItem order[], int& orderSize) {
    if(orderSize == 0) {
        cout << "Your order is empty. There is nothing to remove.\n";
        return;
    }
}

```

```

displayOrder(order, orderSize); // Shahad Al Dhaif . Display the current order

```

```

int removeIndex;

cout << "Enter the number of the item you want to remove: ";

cin >> removeIndex;

```

```

if(removeIndex >= 1 && removeIndex <= orderSize) {
    // Shift the remaining items forward :Fatimah Majid Almar
    for(int i = removeIndex - 1; i < orderSize - 1; ++i) {
        order[i] = order[i + 1];
    }
}

```

```
        orderSize--;  
        cout << "Item removed successfully.\n";  
    } else {  
        cout << "Invalid item number!\n";  
    }  
  
    displayOrder(order, orderSize); // // Shahad Al Dhaif . Display the updated order after  
    removing an item  
}
```

```
// Shahad Al Dhaif
```

```
void displayOrder(OrderItem order[], int orderSize) {  
    cout << "\nCurrent Order:\n";  
    for(int i = 0; i < orderSize; ++i) {  
        cout << i+1 << ". " << order[i].item.name << " - " << order[i].quantity << " x " <<  
        order[i].item.price << " SAR\n";  
    }  
}
```