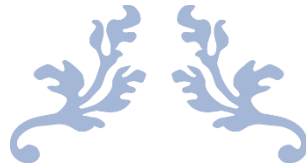


**COMSATS University, Islamabad**

**Islamabad Campus**

**Department of Computer Science**



---

**DATA STRUCTURES AND ALGORITHMS – SP2021**

---



**TOPIC: FOODPANDA MANAGEMENT SYSTEM**

**SUBMITTED TO:**

Dr. Inayat Ur Rehman

**SUBMITTED BY:**

Daoud Hussain (SP21-BCS-102)

# 1. Description of the project (proposal)

The proposed project is about the **CRUD** (Create, Read, Update, delete) based Food Panda Management system in which customers can get orders from the nearby Food Panda. The customers can buy products from the store by first selecting the product category. The available product categories in our stores include **Chicken-Burger (Rs. 200)**, **Andy-wala Burger (Rs.150)**, **Small-Sized-Pizza (Rs. 400)**, **Large-Sized-Pizza (Rs.750)**, **Fries (Rs. 100)**, **One-Liter-Coke (Rs. 120)** and **Shuwarma (Rs.150)**. .

Once the user selects the required product category, the system will check that whether the selected product category stock is available at the shop. If the stock for the selected category is available, the user will be directed to products screen. In case the stock is unavailable for that category, user will be referred to the nearest store where the selected product category is available.

The shortest path to find the nearest store includes traffic rate and the distance to the destination store. The overall cost decides to recommend the nearest store. The system uses **Dijkstra's algorithm** to find the **optimized path** for the nearest store in the food panda network.

The available products are shown to user to buy the desired product. The details of the product include **Order Id, Order name, Order price, Customer name, Customer Mobile number,**).

Food Panda Management system also takes the **feedback** from the user and store in the stack. It also maintains the history of delivered orders to find the sale of shop.

# 2. Data Structures used with purpose

**Linked List** for storage of orders (add, update, cancel, deliver etc).

**Graph** for storing Food Panda costs.

**AVL Trees** for storing delivered orders.

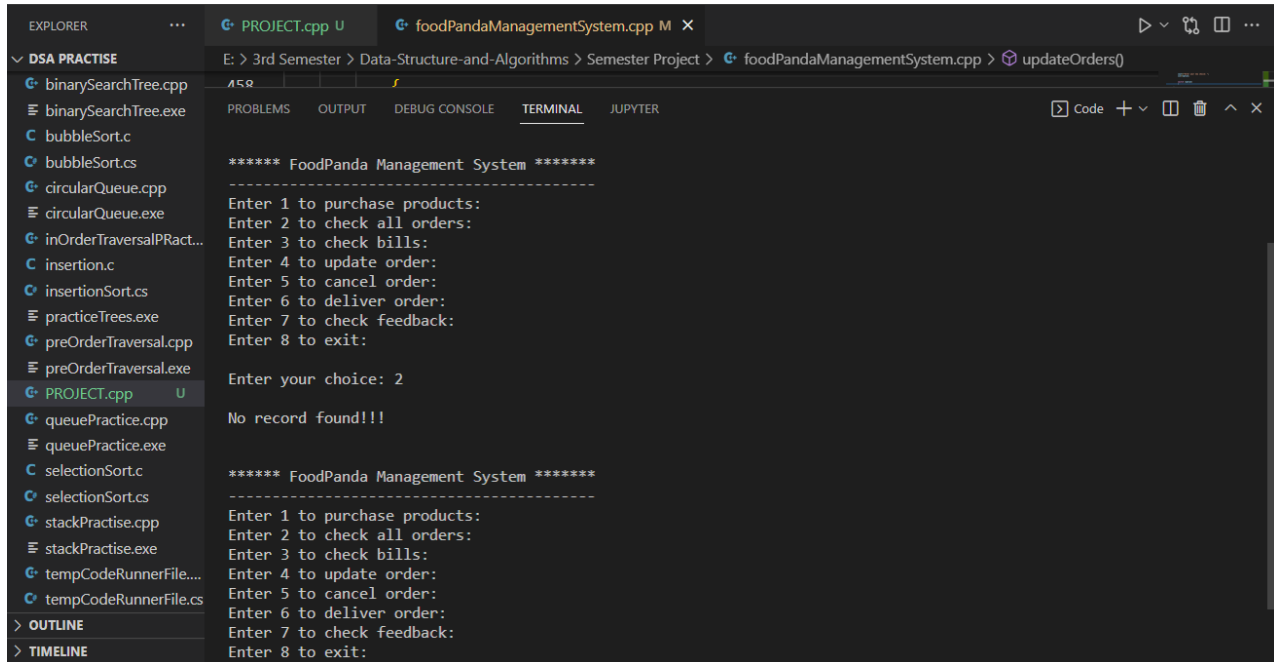
**Stack** for giving and delivering feedback

## *structs used in the code:*

Structure	Purpose
<i>struct Orders</i>	Used to store/retrieve orders.
<i>struct Trees</i>	Used to store information of delivered orders.
<i>struct Stack</i>	Used to give and receive feedbacks.

### 3. Output

#### When Product is out of stock in that market



```
EXPLORER    PROJECT.cpp U    foodPandaManagementSystem.cpp M X
DSA PRACTISE
binarySearchTree.cpp
binarySearchTree.exe
bubbleSort.c
bubbleSort.cs
circularQueue.cpp
circularQueue.exe
inOrderTraversalPract...
insertion.c
insertionSort.cs
practiceTrees.exe
preOrderTraversal.cpp
preOrderTraversal.exe
PROJECT.cpp U
queuePractice.cpp
queuePractice.exe
selectionSort.c
selectionSort.cs
stackPractice.cpp
stackPractice.exe
tempCodeRunnerFile....
tempCodeRunnerFile.cs
OUTLINE
TIMELINE

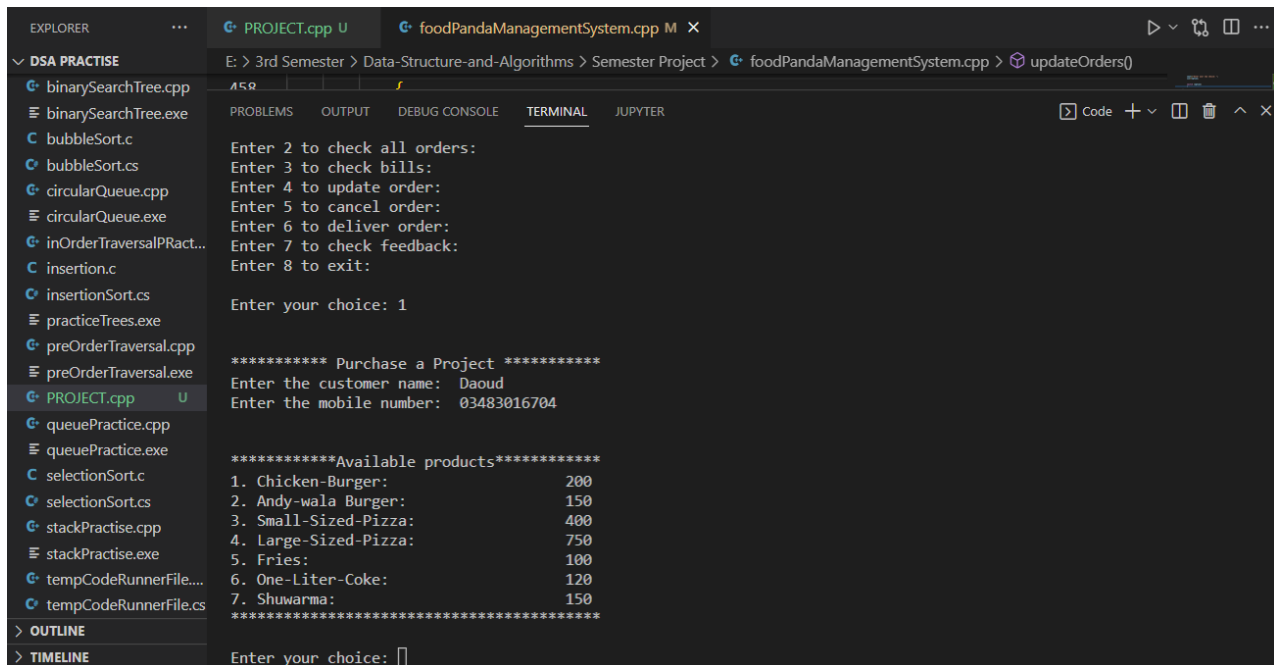
E: > 3rd Semester > Data-Structure-and-Algorithms > Semester Project > foodPandaManagementSystem.cpp > updateOrders()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
***** FoodPanda Management System *****
-----
Enter 1 to purchase products:
Enter 2 to check all orders:
Enter 3 to check bills:
Enter 4 to update order:
Enter 5 to cancel order:
Enter 6 to deliver order:
Enter 7 to check feedback:
Enter 8 to exit:

Enter your choice: 2

No record found!!!

***** FoodPanda Management System *****
-----
Enter 1 to purchase products:
Enter 2 to check all orders:
Enter 3 to check bills:
Enter 4 to update order:
Enter 5 to cancel order:
Enter 6 to deliver order:
Enter 7 to check feedback:
Enter 8 to exit:
```

#### When Product is available in that market



```
EXPLORER    PROJECT.cpp U    foodPandaManagementSystem.cpp M X
DSA PRACTISE
binarySearchTree.cpp
binarySearchTree.exe
bubbleSort.c
bubbleSort.cs
circularQueue.cpp
circularQueue.exe
inOrderTraversalPract...
insertion.c
insertionSort.cs
practiceTrees.exe
preOrderTraversal.cpp
preOrderTraversal.exe
PROJECT.cpp U
queuePractice.cpp
queuePractice.exe
selectionSort.c
selectionSort.cs
stackPractice.cpp
stackPractice.exe
tempCodeRunnerFile....
tempCodeRunnerFile.cs
OUTLINE
TIMELINE

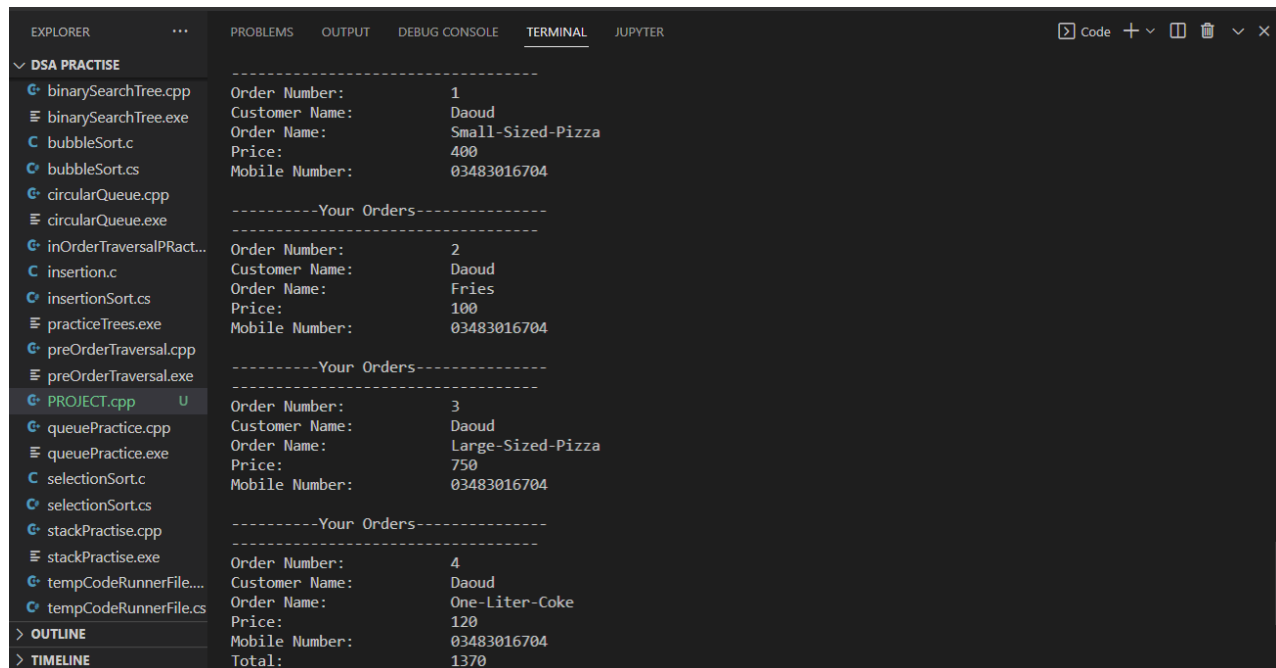
E: > 3rd Semester > Data-Structure-and-Algorithms > Semester Project > foodPandaManagementSystem.cpp > updateOrders()
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Enter 2 to check all orders:
Enter 3 to check bills:
Enter 4 to update order:
Enter 5 to cancel order:
Enter 6 to deliver order:
Enter 7 to check feedback:
Enter 8 to exit:

Enter your choice: 1

***** Purchase a Project *****
Enter the customer name: Daoud
Enter the mobile number: 03483016704

*****Available products*****
1. Chicken-Burger: 200
2. Andy-wala Burger: 150
3. Small-Sized-Pizza: 400
4. Large-Sized-Pizza: 750
5. Fries: 100
6. One-Liter-Coke: 120
7. Shuwarma: 150
*****
Enter your choice: 
```

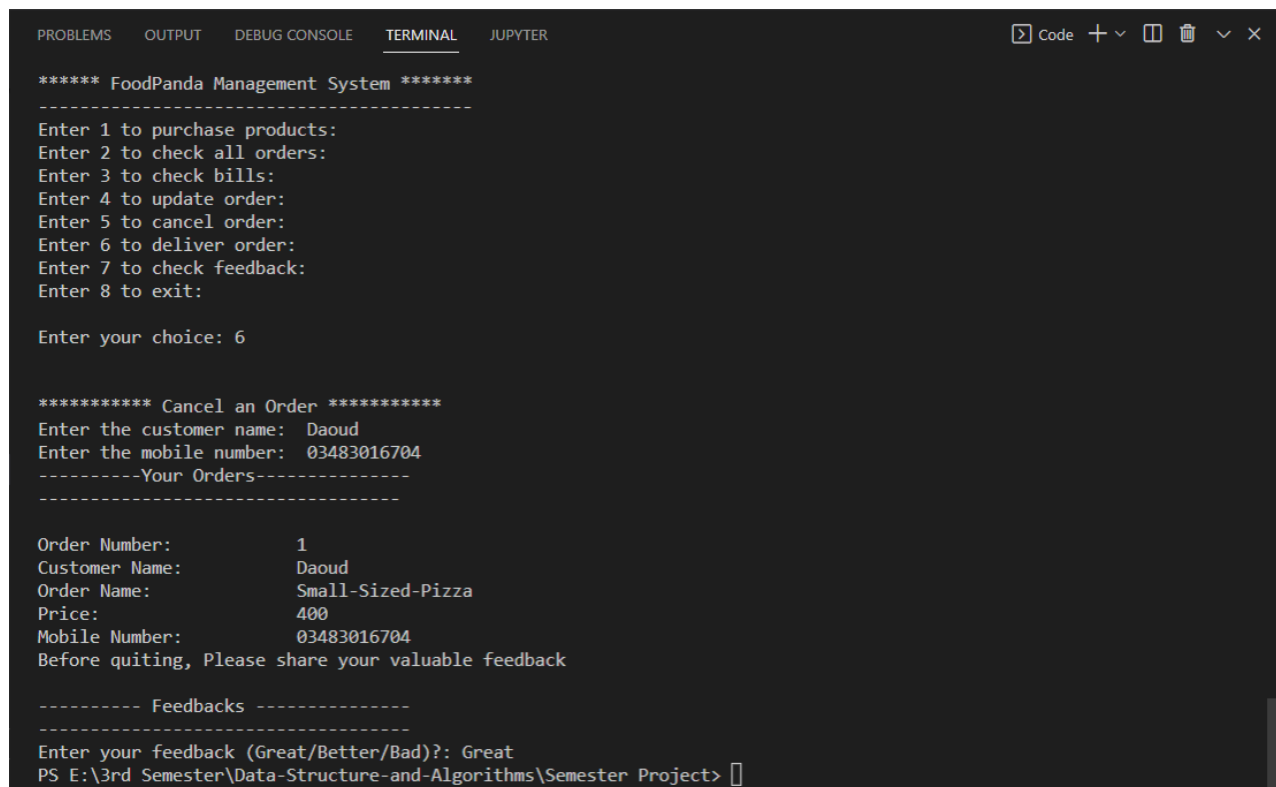
## Bill after purchasing:



The screenshot shows the VS Code Explorer on the left with a file tree under 'DSA PRACTISE'. The file 'PROJECT.cpp' is selected. The Terminal on the right displays the output of the program, showing three orders and their details.

```
-----  
Order Number:      1  
Customer Name:     Daoud  
Order Name:        Small-Sized-Pizza  
Price:             400  
Mobile Number:     03483016704  
  
-----Your Orders-----  
-----  
Order Number:      2  
Customer Name:     Daoud  
Order Name:        Fries  
Price:             100  
Mobile Number:     03483016704  
  
-----Your Orders-----  
-----  
Order Number:      3  
Customer Name:     Daoud  
Order Name:        Large-Sized-Pizza  
Price:             750  
Mobile Number:     03483016704  
  
-----Your Orders-----  
-----  
Order Number:      4  
Customer Name:     Daoud  
Order Name:        One-Liter-Coke  
Price:             120  
Mobile Number:     03483016704  
Total:             1370
```

## Order delivered after Feedback:



The screenshot shows the VS Code Terminal with the output of the program. The user has entered '6' to deliver an order. The program then prompts for customer name and mobile number, and displays the order details. Finally, it prompts for feedback, and the user enters 'Great'.

```
***** FoodPanda Management System *****  
-----  
Enter 1 to purchase products:  
Enter 2 to check all orders:  
Enter 3 to check bills:  
Enter 4 to update order:  
Enter 5 to cancel order:  
Enter 6 to deliver order:  
Enter 7 to check feedback:  
Enter 8 to exit:  
  
Enter your choice: 6  
  
***** Cancel an Order *****  
Enter the customer name: Daoud  
Enter the mobile number: 03483016704  
-----Your Orders-----  
-----  
Order Number:      1  
Customer Name:     Daoud  
Order Name:        Small-Sized-Pizza  
Price:             400  
Mobile Number:     03483016704  
Before quitting, Please share your valuable feedback  
  
----- Feedbacks -----  
-----  
Enter your feedback (Great/Better/Bad)? : Great  
PS E:\3rd Semester\Data-Structure-and-Algorithms\Semester Project>
```

## Exit

EXPLORER

...

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

JUPYTER

Code + - [ ] [X] [X] [X] [X]

DSA PRACTISE

binarySearchTree.cpp

binarySearchTree.exe

bubbleSort.c

bubbleSort.cs

circularQueue.cpp

circularQueue.exe

inOrderTraversalPRACTISE.cpp

insertion.c

insertionSort.cs

practiceTrees.exe

preOrderTraversal.cpp

preOrderTraversal.exe

PROJECT.cpp

queuePractice.cpp

queuePractice.exe

selectionSort.c

selectionSort.cs

stackPractise.cpp

stackPractise.exe

tempCodeRunnerFile....

tempCodeRunnerFile.cs

OUTLINE

TIMELINE

1. Chicken-Burger: 200

2. Andy-wala Burger: 150

3. Small-Sized-Pizza: 400

4. Large-Sized-Pizza: 750

5. Fries: 100

6. One-Liter-Coke: 120

7. Shuwarma: 150

\*\*\*\*\*

Enter your choice: 2

Andy-wala-Burger is purchased Successfully

\*\*\*\*\* FoodPanda Management System \*\*\*\*\*

-----

Enter 1 to purchase products:

Enter 2 to check all orders:

Enter 3 to check bills:

Enter 4 to update order:

Enter 5 to cancel order:

Enter 6 to deliver order:

Enter 7 to check feedback:

Enter 8 to exit:

Enter your choice: 8

Thank you for visiting our bakery!!

Have a great day Sir!!

PS E:\3rd Semester\Data-Structure-and-Algorithms\Semester Project>