

ToGoEat

COMSC-210-5221

By: Jiefeng Yang, Jason Dao, Max Raffield

Project Purpose

Use data structures to create a simulation of
managing + **ordering** from a restaurant.

- **Edit restaurant menu:** pricing, cooking time, revenue.
- **Track order data:** customer name, their items and total.
- Program **organizes orders** based on time + chef availability

Features and Functionality

- **Manager-side: modification of restaurant** (adding/removing items from a menu)
- **Customer-side: place orders** and observe restaurant
- Paying and cancellation functionality included to complete orders.
- Multiple chefs cook orders which finish depending on preparation time.
- Revenue is calculated based on orders per day as well as the individual items.
- **All progress + changes are saved** between runs. (stored as a .txt file).

Manager-side

[Name Restaurant] - gives the restaurant a name.

[Add Item] - prompts input for item details and adds item to the menu.

[Remove Item] - removes an item of the specified ID from the menu.

[Print Current Menu] - prints the current menu, item IDs, names and prices.

[Hire Chef] - add another chef with enough revenue to speed up operations

[Exit] - exits the manager-side.

Customer-side

[Observe] - skips the creation of an order.

[Make Order] - prompts name, displays menu and gives several selection options.

[Add Item to Order] - prompts an item ID and adds matching item to order.

[Clear All Items] - removes all items from order.

[Print Order] - displays items currently in the order as well as the total.

[Finish] - prompts payment and then adds order to the wait queue.

[Cancel] - cancels order.

Implementing our own Data Structures

Queue - *a FIFO (first in, first out) data structure.*

Program uses a Queue as a waitlist to hold customer orders.

Priority Queue - *the order of the queue items depends on their priority.*

Program uses a Priority Queue for the eventQueue in lo-to-hi priority order, used to track end-of-service times.

Dynamic Array - *an array that has dynamic size.*

The program uses a Dynamic Array to hold chef's data.

Application Demonstration

Test Scenarios Checklist

- Selecting Manager/Customer Sides (✓)
- Input Validation for User Input (✓)
- Editing Menu Items (✓)
- Displaying Menus (✓)
- Hiring of Chefs (✓)
- Modifying Menu.txt File (✓)
- Making Orders (✓)
- Adding/Removing/Clearing Order Items (✓)
- Displaying Order With Total (✓)
- Finishing/Cancelling Orders and Payment (✓)
- Cooking/Wait Queues (✓)

We tested our data structures with
test drivers.

		Test Cases			
Test Case #	Test Case Description	Test Data	Expected Result	Actual Result	Pass/Fail
1	Choose Client Mode	Enter 1 for Client Mode	Goes to Client Mode	Goes to Client Mode	Pass
2	Choose Manager Mode	Enter 2 for Manager Mode	Goes to Manager Mode	Goes to Manager Mode	Pass
3	Choose Invalid Input	Enter 3, abc for Manager Mode	Shows Error Info and Prompts Re-Input	Shows Error Info and Prompts Re-Input	Pass
4	Naming Restaurant	Enter a string	Names Restaurant	Names Restaurant	Pass
5	Select Exit	Enter 0 to Exit	Exits	Exits	Pass
6	Select Add Item	Enter 1 to Add Menu Item	Asks for Item Details	Asks for Item Details	Pass
7	Input Item Details	Enter Name, Price, Preparation Time	Adds Item to Menu	Adds Item to Menu	Pass
8	Select Remove Item	Enter 2 to Remove Menu Item	Asks for Item to Remove	Asks for Item to Remove	Pass
9	Input Item ID to Remove	Enter an Integer	Removes Item at Specified Integer	Removes Item at Specified Integer	Pass
10	Select Print Current Menu	Enter 3 to Print Current Menu	Prints Menu	Prints Menu	Pass
11	Pick Invalid Manager Action	Enter a String	Prompts Re-Input	Prompts Re-Input	Pass
12	Ignore Making Order	Enter I to Ignore	Skips Making an Order	Skips Making an Order	Pass
13	Start an Order	Enter M to Start Order	Asks for Name	Asks for Name	Pass
14	Naming Customer	Enter a String	Sets Customer Name and Displays Menu	Sets Customer Name and Displays Menu	Pass
15	Select Invalid Option	Enter a String	Prompts Re-Input	Prompts Re-Input	Pass
16	Select Add Item to Order	Enter 1 to Add Order Item	Asks for Item ID	Asks for Item ID	Pass
17	Add Item to Order	Enter ID of Menu Item	Adds Item to Order	Adds Item to Order	Pass
18	Enter Invalid Item ID	Enter Invalid Item ID	Prompts Re-Input	Prompts Re-Input	Pass
19	Select Clear Items	Enter 2 to Clear Items	Clears Items	Clears Items	Pass
20	Select Print Order	Enter 3 to Print Order	Prints Order Details	Prints Order Details	Pass
21	Select Cancel	Enter 5 to Cancel Order	Cancels Order	Cancels Order	Pass
22	Select Finish	Enter 4 to Finish Order	Asks for Payment	Asks for Payment	Pass
23	Select Pay	Enter 1 to Pay	Finishes Order and Prints Receipt	Finishes Order and Prints Receipt	Pass
24	Select Cancel Payment	Enter 2 to Cancel Payment	Cancels Order	Cancels Order	Pass
25	Enter Invalid Payment Option	Enter a String	Prompts Re-Input	Prompts Re-Input	Pass
26	Order Completion	Wait Until End-Of-Service Time	Order Finishes	Order Finishes	Pass
27	Checking Program Summary	Add 2 Orders, End Program	Displays 2 Orders and Totals	Displays 2 Orders and Totals	Pass
28	Check End Service Time	Verify End Service Time of Order (Main)	Equal to End Service Time of Order	Equal to End Service Time of Order	Pass
29	Server List Includes Customer Name/Order ID	Verify Customer Name/Order ID of Order (Main)	Equal to Customer Name and Order ID	Equal to Customer Name and Order ID	Pass
30	Check Waitlist	Add Order While Chefs Busy	New Order Added to Waitlist	New Order Added to Waitlist	Pass
31	Verify Ending Client Mode	Add Orders and Ignore	Client Mode Ends After Completing Old Orders	Client Mode Ends After Completing Old Orders	Pass
32	Verify Changing menu.txt	Use Manager Mode to Modify Menu	The .txt File Includes Changes	The .txt File Includes Changes	Pass
33	Restaurant name is present on first run	Not present	Prompt for input and save to file and continue reading	Prompt for input and save to file and continue reading	Pass
34	Restaurant name is present on first run	Present	Save to file and continue reading	Save to file and continue reading	Pass
34	Request to hire chef	Enter 4	Show hire chef menu	Show hire chef menu	Pass
35	Request to hire chef	Enter string	Print error and return to previous menu	Print error and return to previous menu	Pass
36	Request to hire chef	Enter 4 (not enough money)	Cancel and return to previous menu	Cancel and return to previous menu	Pass
37	Hire chef	Enter 1	Chef is incremented and save to config file	Chef is incremented and save to config file	Pass
38	Hire chef	Enter string	Print error and return to previous menu	Print error and return to previous menu	Pass
39	Hire chef	Enter 2	Cancel and return to previous menu	Cancel and return to previous menu	Pass

Obstacles and Limitations

- Overall design changed over the course of production due to **shifting ideas**.
- Renamed variables would create **inconsistencies across files**.
- **Version Control:** Team would sometimes accidentally work on different versions of files.
- Removed some ideas that were too complicated given **limited remaining time**.

Future Plan Ideas

- Could be further developed into an **app**!
- Additional features and aesthetic updates could be implemented.
- Delivery system that calculates delivery time based on distance.
- Adding randomization to preparation time.

Thank You For Listening!