Cafetaria Management

Team members :

Sayed.Muskan (Teamlead) -24KB1A05GS

Sd.Esha parwaz -24KB1A05KH

Y.Bhuvaneswari. -24KB1A05PB

Group &Section - CSE-D

Introduction

The cafeteria system is simple yet efficient solution for managing food ordering and cafetaria operations .This system is designed to streamline the process of ordering through managing menu items and tracking user balances

Objective

User-Friendly Design:

Easy navigation for customers and administrators.

Efficient Management:

Fast food ordering and inventory tracking.

Optimal Performance:

Utilize data structures and algorithms for speed.

Why C & DSA ?

Efficiency

Fast execution and performance
C's efficiency comes from its compiled nature and minimal overhead, making it ideal for systems programming and high-performance applications.

Portability

Allows the code to be compiled on various platforms with minimal modifications.
Easy to port across different platforms

Efficient data management

Data structures and algorithms for
Efficient data management that enables organized storage and retrieval of menu items, user data, and orders, ensuring fast and accurate service.

Problem solving

DSA enables problem-solving by developing efficient algorithms to tackle complex issues, such as optimizing menu item searches, managing inventory, and streamlining order processing.

Algorithm

Step 1: Initialize Program
Define MenuItem and User structures using linked lists.
Set global variables for menu and user linked list heads and IDs.

Step 2: Load Data
Call loadMenu() to read menu items from "menu.txt".
If file doesn't exist, create a new empty "menu.txt".
Call loadUsers() to read user accounts from "user.txt".
If file doesn't exist, create a new empty "user.txt".

Step 3: Display Main Menu
Repeatedly show main menu options:
1. Login (User)   2. Login (Admin)
3. View Menu   4. Take Order
5. View Balance   6. Logout
7. Add Menu Item (Admin - restricted)
8. Delete Menu Item (Admin - restricted)
9. Update Menu Item (Admin - restricted)
10. Add User (Admin - restricted)
11. Exit

Step 4: Handle User Choices
Case 1: User Login
Prompt for card number.
Find user in linked list (findUserByCard()).
If valid, login success; else, fail.

Case 2: Admin Login
Prompt for admin PIN.
If correct:
Display Admin Menu:
1. View Menu   2. Add Menu Item
3. Delete Menu Item   4. Update Menu Item
5. Add User   6. Back to Main Menu
Perform selected admin tasks.

Case 3: View Menu
Display all menu items with ID, name, price, availability.

Case 4: Take Order
Must be logged in as user.
Show menu.
Take multiple item orders:
Check item availability.
Update item stock.
Calculate total bill.
Deduct from user balance if sufficient.

Case 5: View Balance
Show logged-in user's current balance.

Case 6: Logout
Log out the current user.

Cases 7-10: Admin Only Operations
If not logged in as Admin, display error message.
Case 0: Exit
Exit loop.

Step 5: Admin Menu Options (after Admin Login)
View Menu: Show all menu items.

Add Menu Item:
Ask for item details (name, price, availability).
Add new node to menu linked list.
Save to file.

Delete Menu Item:
Ask for ID.
Find and remove menu item from linked list.
Save to file.

Update Menu Item:
Ask for ID.
Allow updating name, price, availability.
Save to file.

Add User:
Enter username and card number.
Assign initial balance.
Save to file.

Step 6: Memory Management
Before program ends:
Free memory used by MenuItem linked list.
Free memory used by User linked list

Lessons Learnt

01 Efficient Coding and Problem-Solving*
- Writing optimized code that minimizes resource usage and maximizes performance.
- Breaking down complex problems into manageable parts and solving them efficiently.

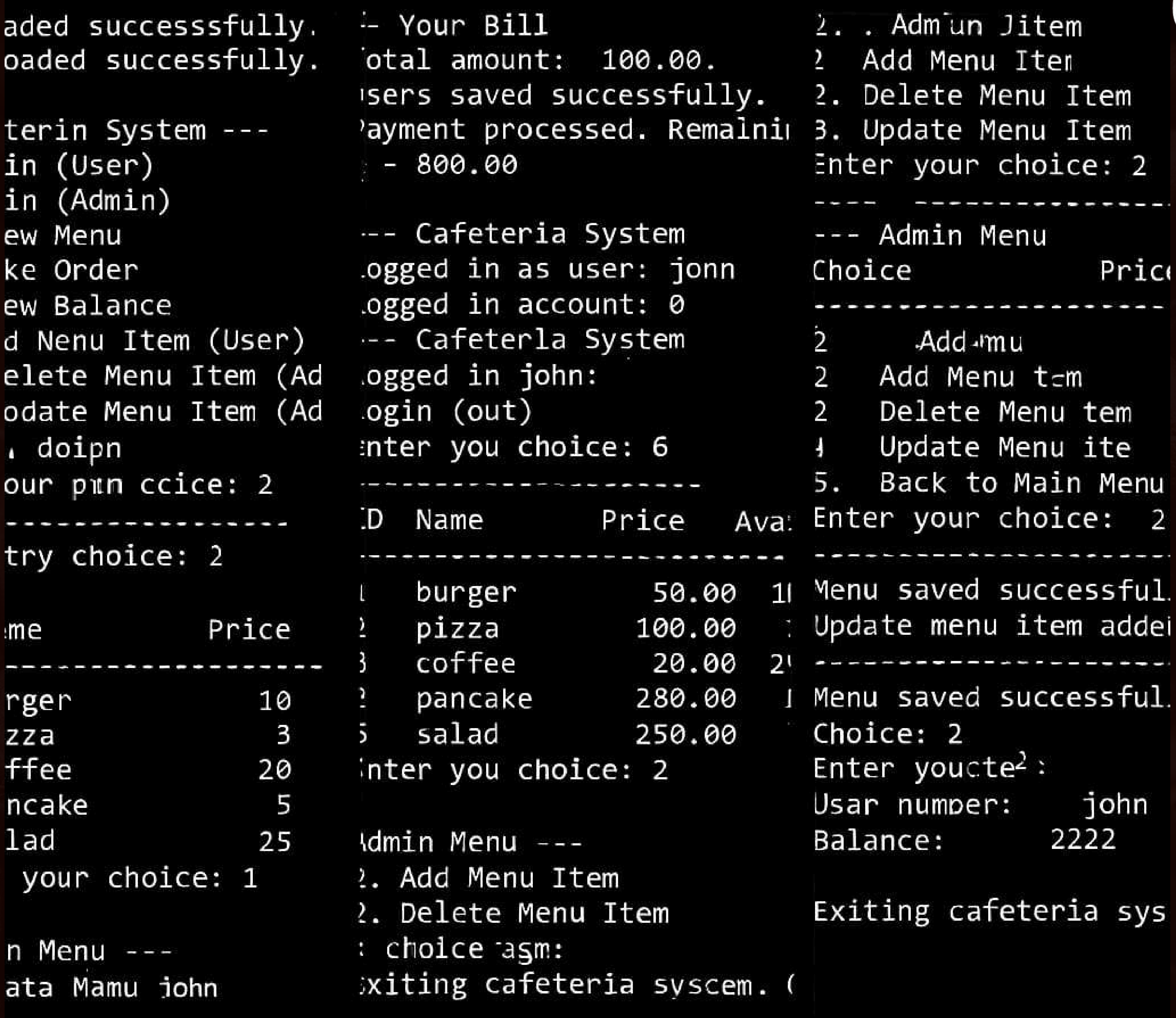
02 Effective Data Management*
- Organizing and structuring data for efficient retrieval and manipulation.
- Choosing the right data structures to suit the problem's requirements.

03 Scalable System Design*
- Designing systems that can handle increased traffic, data, or user demands.
- Building systems that are flexible and adaptable to changing requirements.

04 Practical Application of Theoretical Concepts*
- Applying theoretical knowledge of C and DSA to real-world problems.
- Translating concepts into working solutions that meet practical needs

Cafeteria Management

Output



```
aded successsfully.
oaded successfully.

terin System ---
in (User)
in (Admin)
ew Menu
ke Order
ew Balance
d Nenu Item (User)
elete Menu Item (Ad
odate Menu Item (Ad
, doipn
our pıın ccice: 2
-----------------
try choice: 2

ıme            Price
------------------
rger              10
zza                3
ffee              20
ncake              5
lad               25
 your choice: 1

n Menu ---
ata Mamu john
```

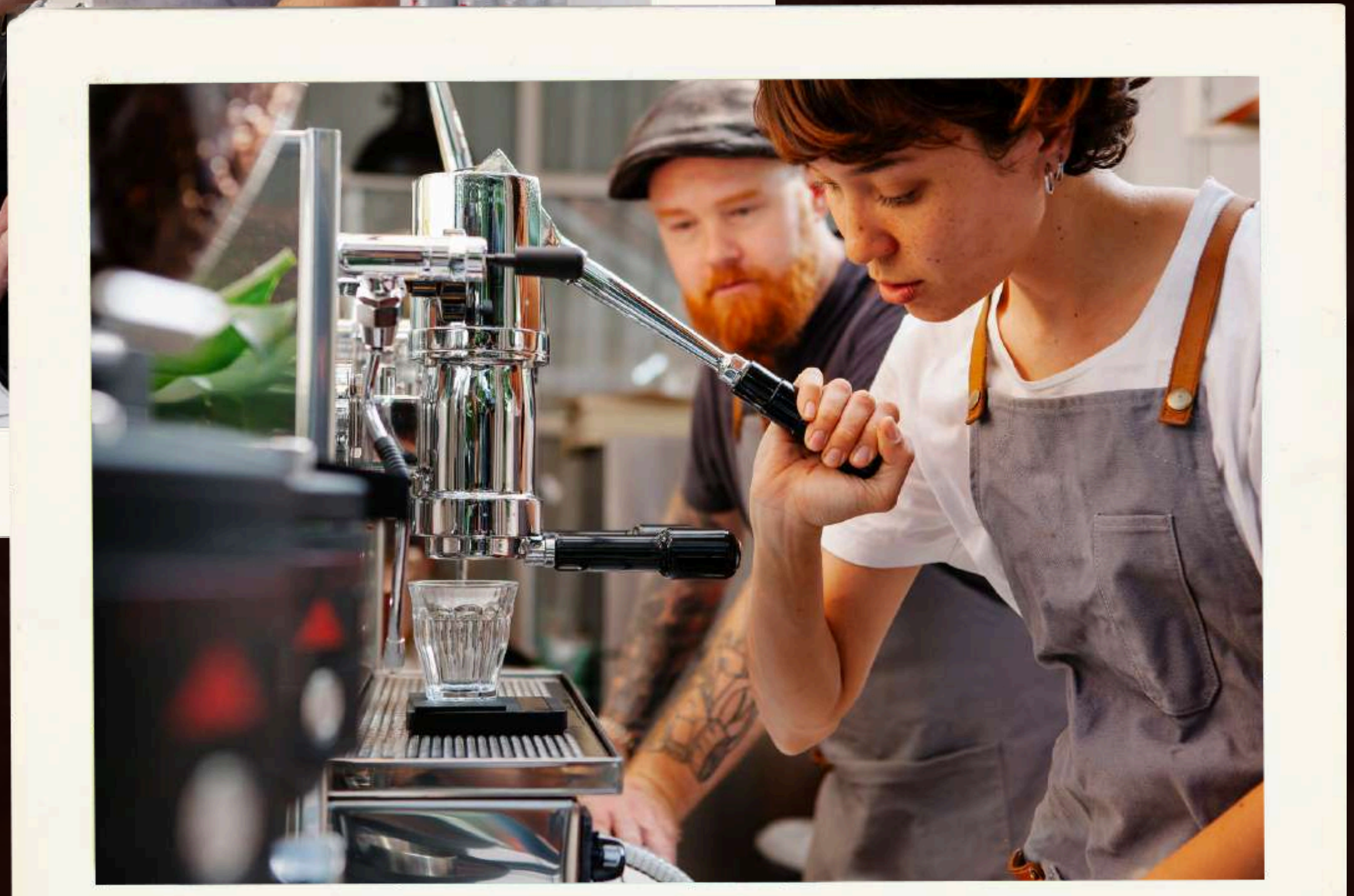
```
-- Your Bill
'otal amount:  100.00.
ısers saved successfully.
'ayment processed. Remainii
 - 800.00

--- Cafeteria System
.ogged in as user: jonn
.ogged in account: 0
--- Cafeterla System
.ogged in john:
.ogin (out)
:nter you choice: 6
--------------------
:D  Name        Price   Ava:
-------------------------
l   burger        50.00  1(
2   pizza        100.00   1
3   coffee        20.00  2!
2   pancake      280.00   I
5   salad        250.00
:nter you choice: 2

\dmin Menu ---
2. Add Menu Item
2. Delete Menu Item
: choice asm:
:xiting cafeteria syscem. (
```

```
2. . Admun Jitem
2  Add Menu Itern
2. Delete Menu Item
3. Update Menu Item
Enter your choice: 2
----  ---------------
--- Admin Menu
Choice            Price
--------------------
2      Add mu
2    Add Menu tem
2    Delete Menu tem
4    Update Menu ite
5.   Back to Main Menu
Enter your choice:  2
--------------------
Menu saved successful.
Update menu item addei
--------------------
Menu saved successful.
Choice: 2
Enter youcte:
Usar numper:    john
Balance:      2222

Exiting cafeteria sys
```



Conclusion

The Cafeteria Management System project has successfully demonstrated the development of an efficient and user-friendly system for managing cafeteria operations. By leveraging C programming language and data structures and algorithms, the system provides optimal performance, scalability, and reliability.

Thank You