N.B.K.R.I.S.T
VIDYANAGAR
AUTONOMUS

SBI CREDIT CARD PROCESSING APPLICATION

SBI CREDIT CARD PROCESSING APPLICATION SYSTEM USING QUEUE IN C

UNDER GUIDENCE

PRASANTH SIR
SURESH GUPTHA SIR

PRESENTED BY

V.BHANUPRAKESHREDDY (24KB1A30C5)

CH.MAHESH (24KB1A3017)

J.SAI KRISHNA (24KB1A3035)

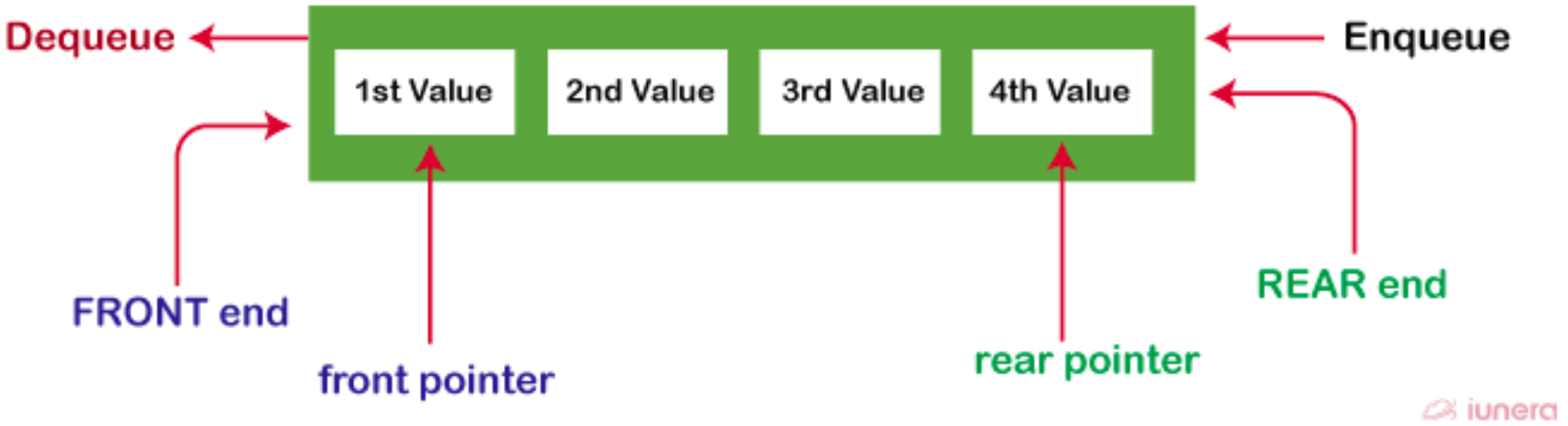
G.SRI HARI (24KB1A3029)

OBJECTIVE

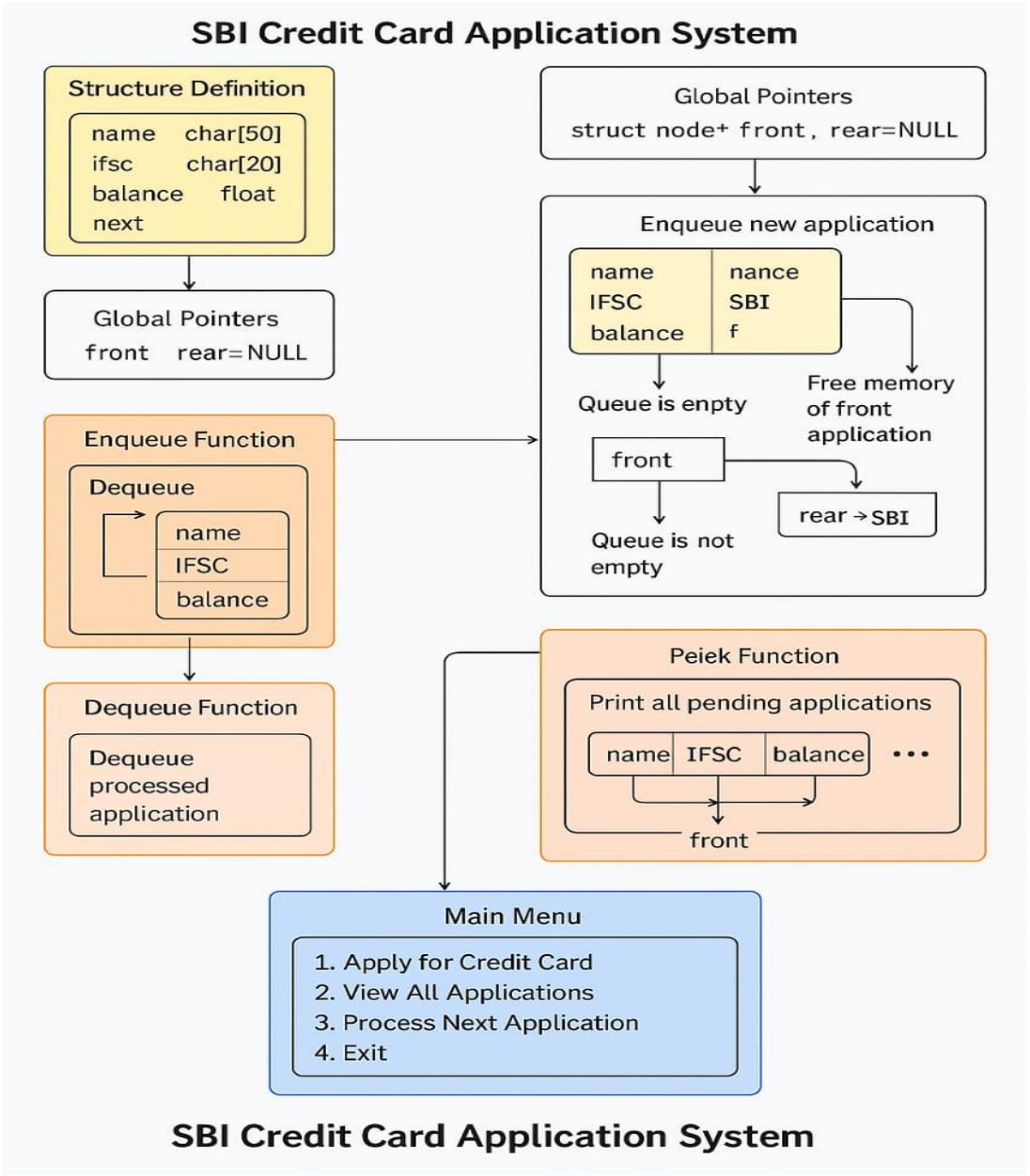
This project simulates a real-world credit card application processing system using data structures. It uses a queue-based approach to handle multiple applications in the order they are received. Linked list concepts are applied for dynamic memory allocation and flexibility in managing the queue. Users can apply, view, and process credit card applications efficiently through a command-line menu interface. The system helps reinforce understanding of queue operations like enqueue, dequeue, and peek using C programming.

QUEUE

- A Queue Data Structure is a fundamental concept in computer science used for storing and managing data in a specific order.
- It follows the principle of "First in, First out" (FIFO), where the first element added to the queue is the first one to be removed.



Diagramatic view of the application

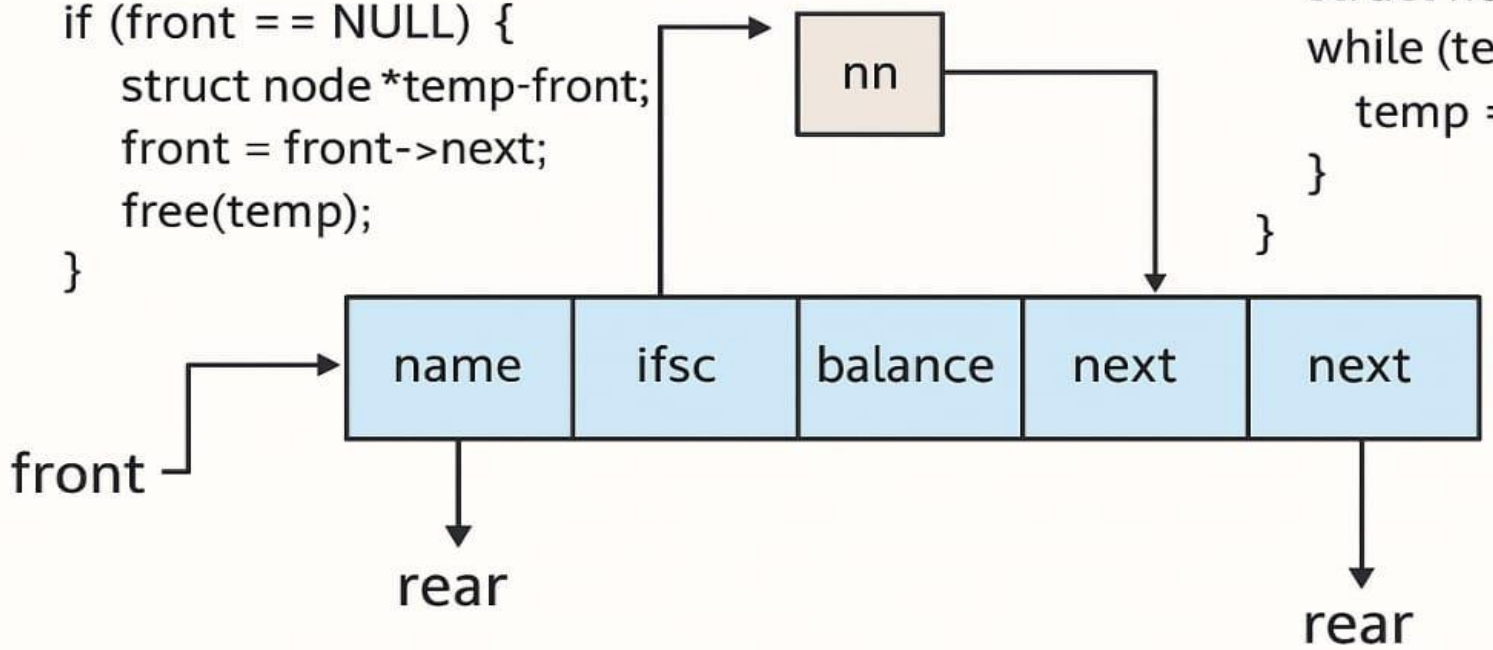


```
struct node* front =
NULL, rear = NULL;
```

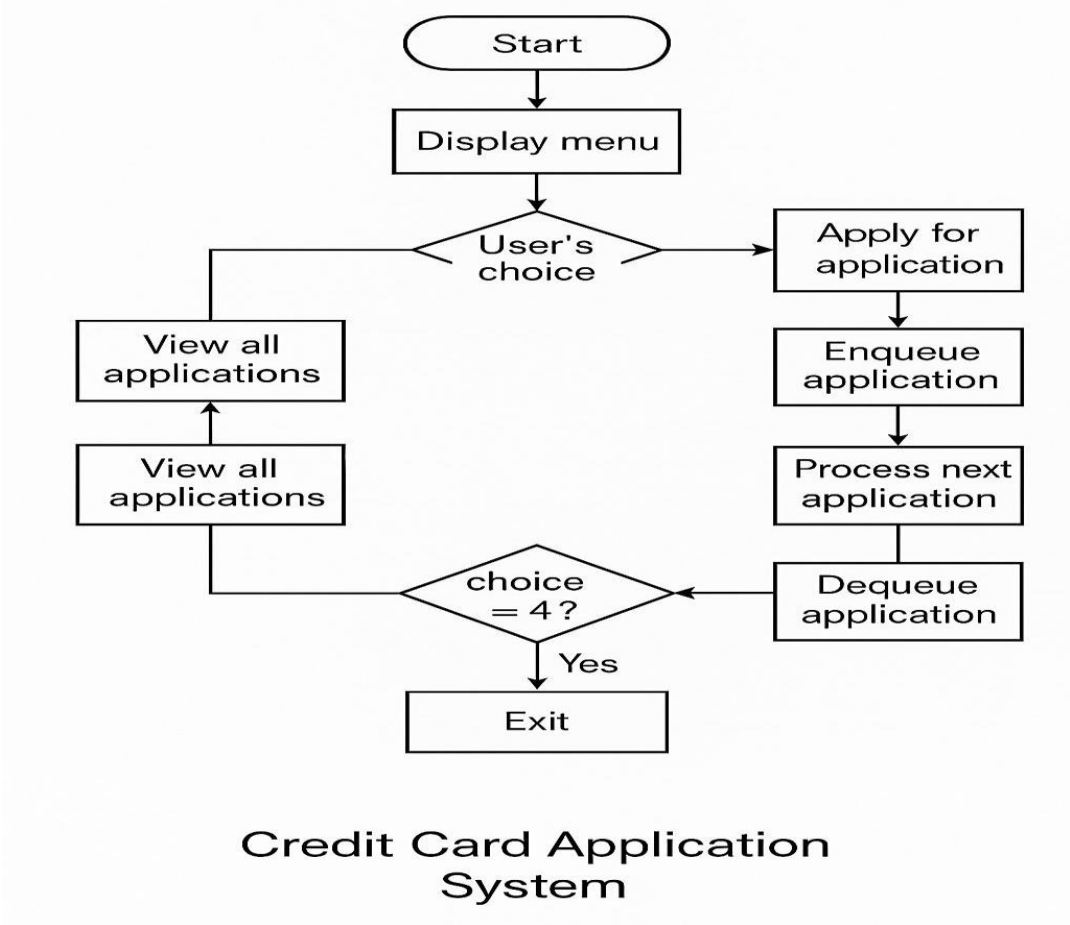
```
void enqueue(char name[], char ifsc [], float balance) {
        rear->next = nn;
        rear = nn;
}
```

```
void dequeue()
    if (front == NULL) {
        struct node *temp-front;
        front = front->next;
        free(temp);
    }
```

```
void peek() {
    struct node*temp = front;
    while (temp != NULL) {
        temp = temp->next;
    }
}
```



FLOW CHART

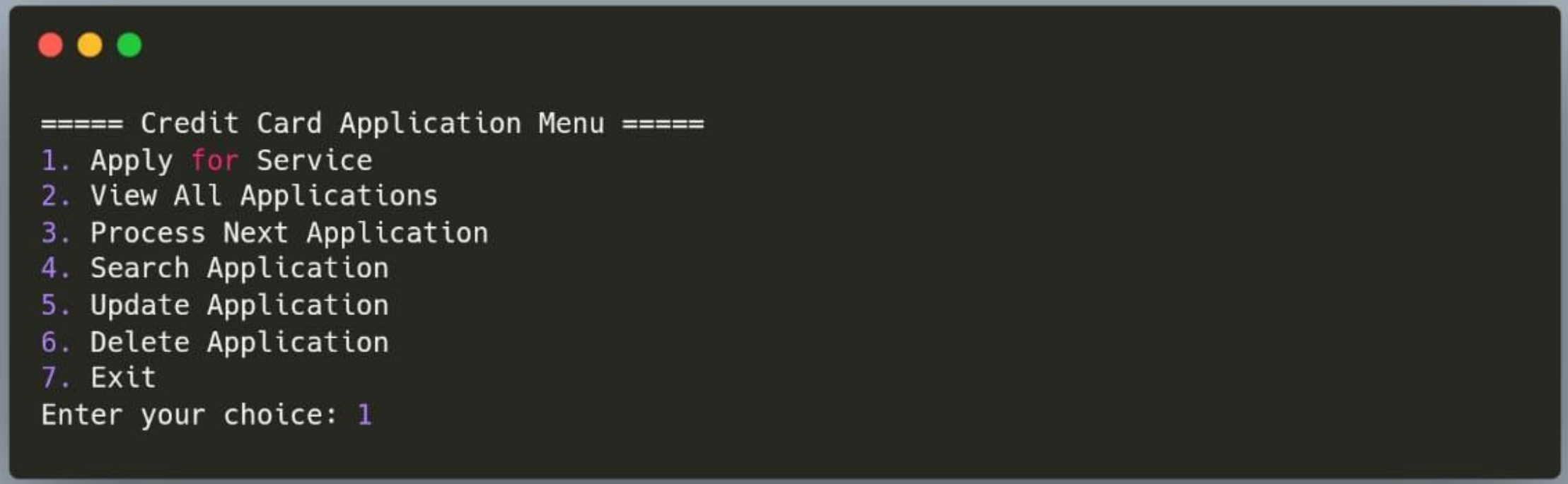


Credit Card Application System

SOURCE LINK

https://github.com/maheshchallagiri/SBI-banking-application-process-/blob/main/main.c#L1

Sample output



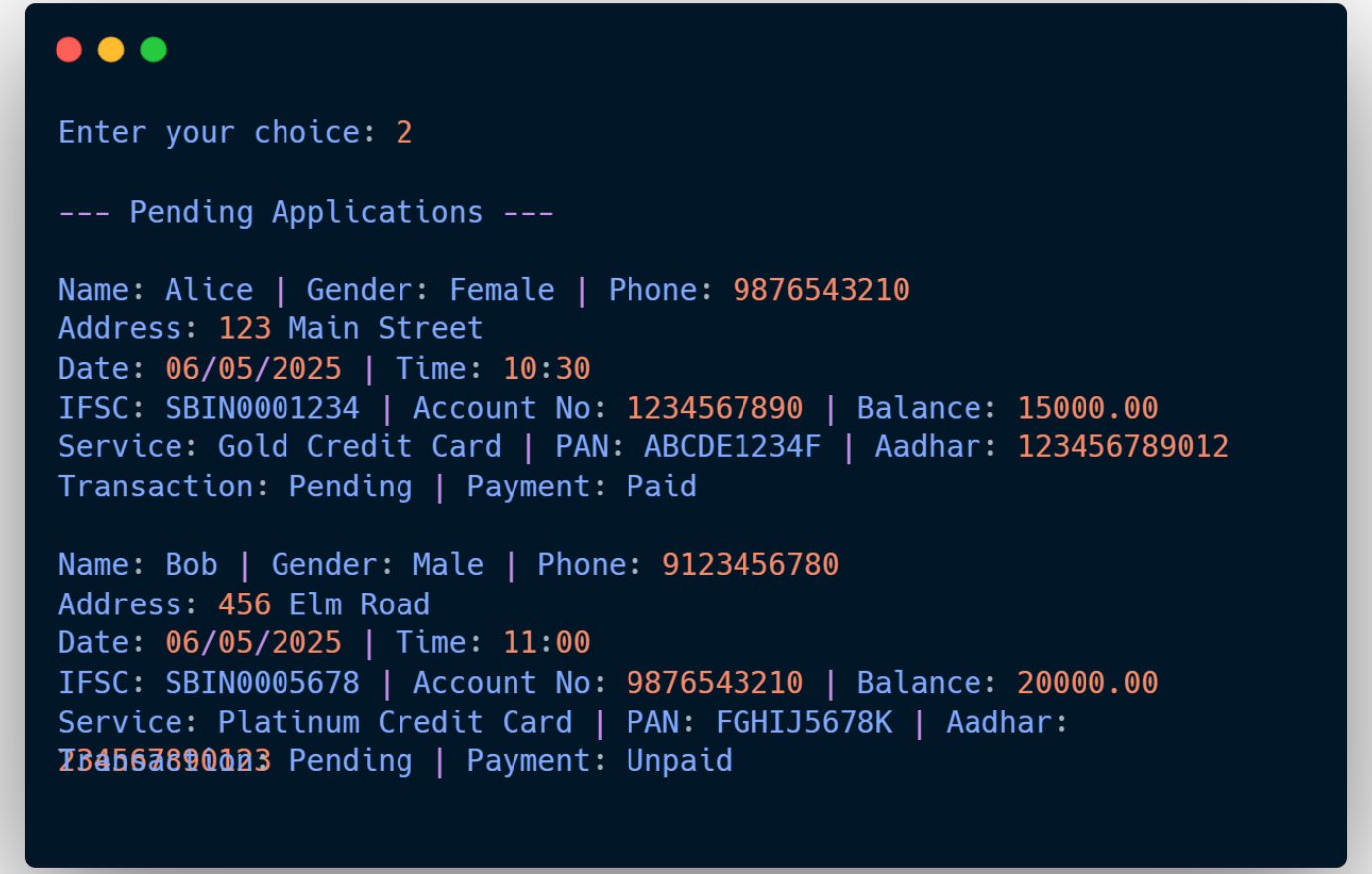
(User applies for a service)

```
Enter Name: Alice
Enter Gender: Female
Enter Address: 123 Main Street
Enter Date (dd/mm/yyyy): 06/05/2025
Enter Time (hh:mm): 10:30
Enter IFSC Code: SBIN0001234
Enter Account Number: 1234567890
Enter Balance: 15000
Enter Service Type: Gold Credit Card
Enter Phone Number: 9876543210
Enter Transaction Status: Pending
Enter PAN Number: ABCDE1234F
Enter Aadhar Number: 123456789012
Enter Payment Status (Paid/Unpaid): Paid
Application submitted successfully for Alice.
```

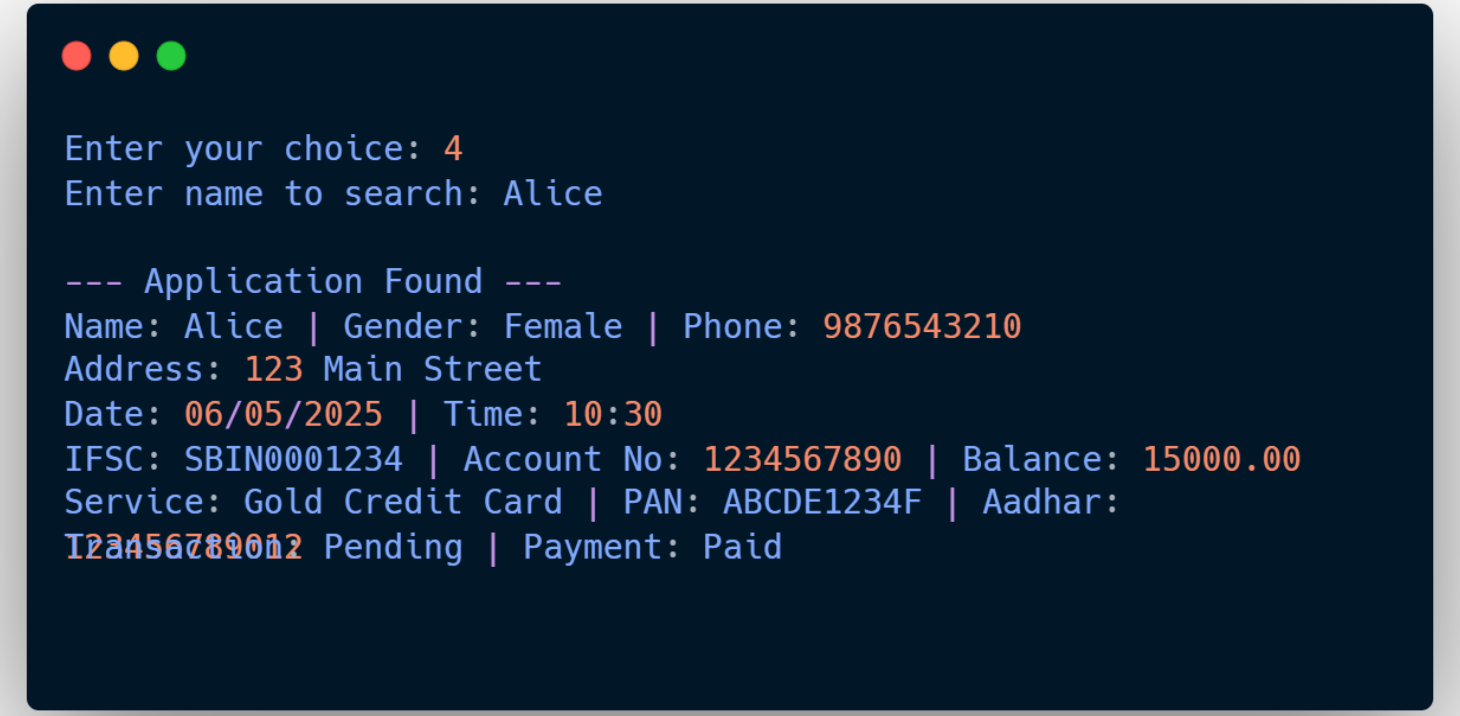
User adds another application:

```
Enter your choice: 1
Enter Name: Bob
Enter Gender: Male
Enter Address: 456 Elm Road
Enter Date (dd/mm/yyyy): 06/05/2025
Enter Time (hh:mm): 11:00
Enter IFSC Code: SBIN0005678
Enter Account Number: 9876543210
Enter Balance: 20000
Enter Service Type: Platinum Credit Card
Enter Phone Number: 9123456780
Enter Transaction Status: Pending
Enter PAN Number: FGHIJ5678K
Enter Aadhar Number: 234567890123
Enter Payment Status (Paid/Unpaid): Unpaid
Application submitted successfully for Bob.
```

User views all applications:



User searches for Alice:

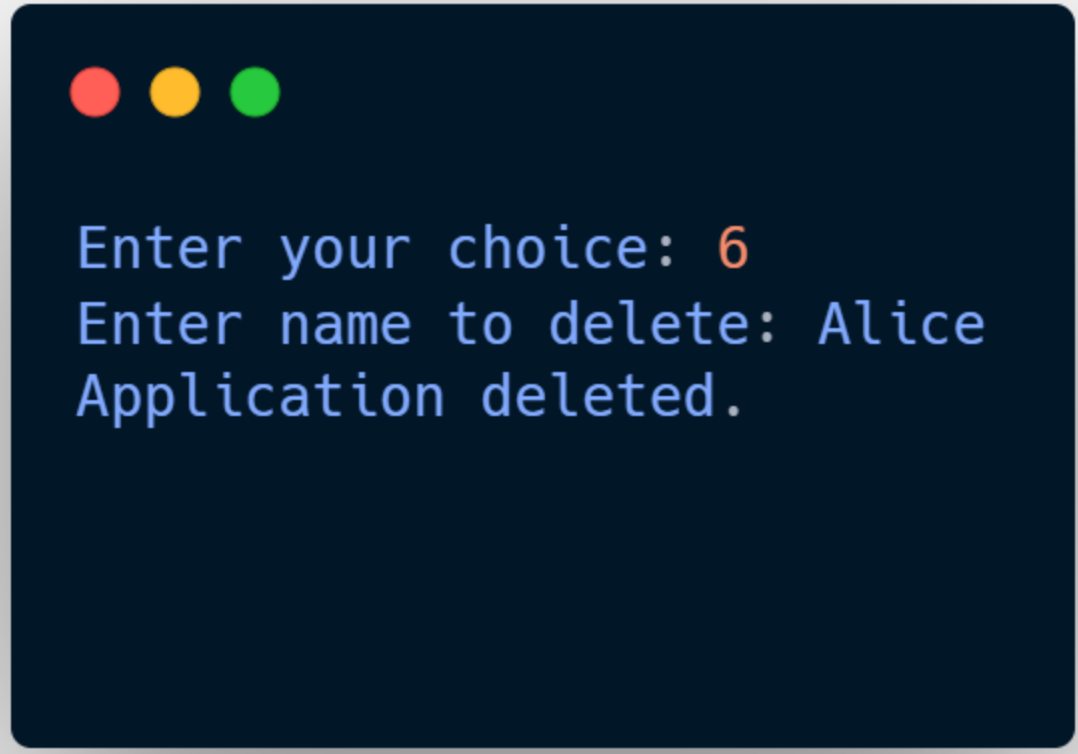


User updates Bob’s application:

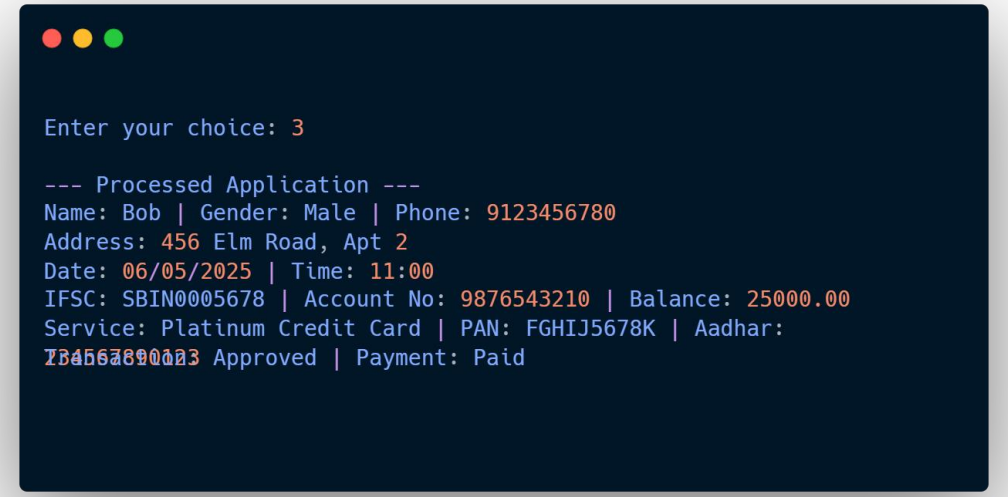
```
Enter your choice: 5
Enter name to update: Bob

--- Updating Application ---
Enter Name: Bob
Enter Gender: Male
Enter Address: 456 Elm Road, Apt 2
Enter Date (dd/mm/yyyy): 06/05/2025
Enter Time (hh:mm): 11:00
Enter IFSC Code: SBIN0005678
Enter Account Number: 9876543210
Enter Balance: 25000
Enter Service Type: Platinum Credit
Enter Phone Number: 9123456780
Enter Transaction Status: Approved
Enter PAN Number: FGHIJ5678K
Enter Aadhar Number: 234567890123
Enter Payment Status (Paid/Unpaid):
Application updated.
```

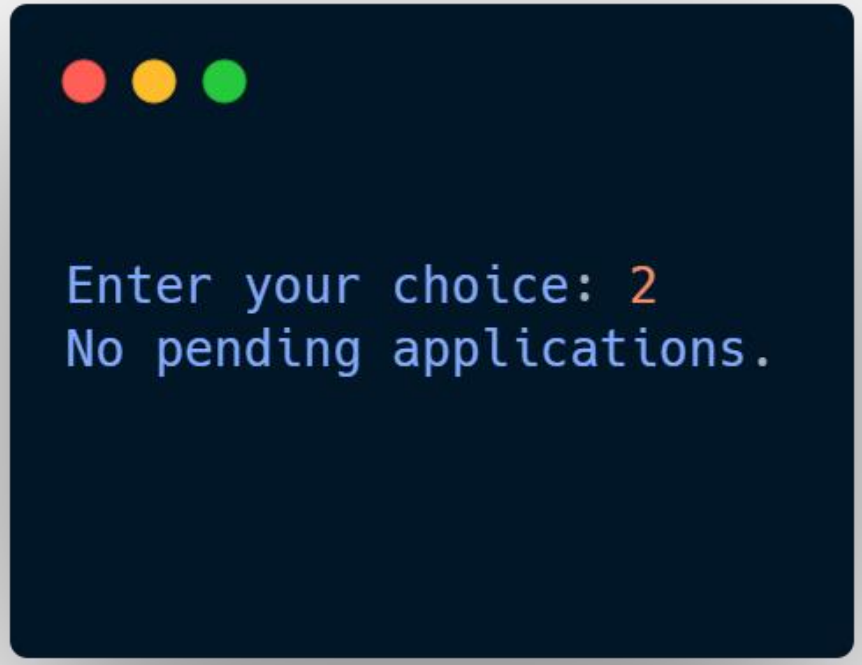
User deletes Alice’s application:



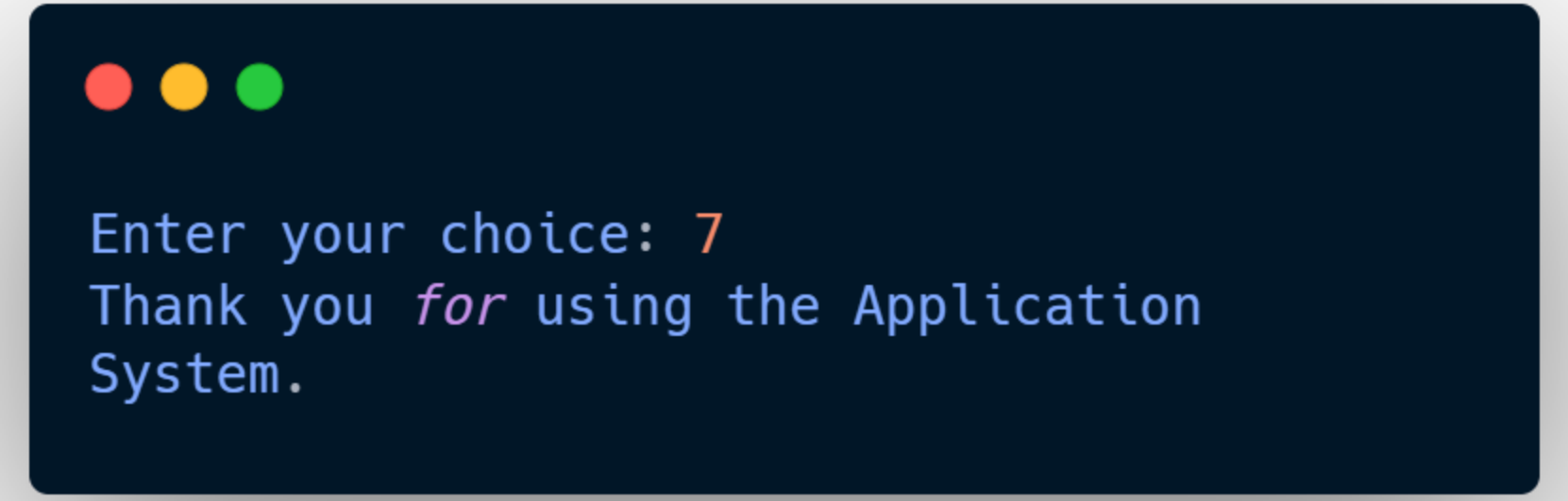
User processes the next application (Bob):



User tries to view applications again:



User exits the program:



✅ Conclusion

- In this project, we created a simple system to manage credit card applications using the concept of **queues** in C language.
- Users can apply for a card, view their application, search by name, update or delete it, and process applications one by one in the order they came.
- We used a **linked list** to make the queue work well even when applications are added or removed.
- This project helped us understand **how data structures like queues** work in real programs. We also practiced **functions, pointers, and user input** in C.


THANK YOU