**LAB 4 - Circular Queue**

**Name : Abhinav Sanjay**

**USN : 1BM23CS009**

#include <stdio.h>

#include <stdlib.h>

#define SIZE 3

int queue[SIZE];

int front = -1;

int rear = -1;

void insert (int value) {

if ((front == 0 && rear == SIZE - 1) || (rear == (front - 1) % (SIZE - 1))) {

printf("Queue is Full\n");

return;

}

else if (front == -1) {

front = rear = 0;

queue[rear] = value;

}

else if (rear == SIZE - 1 && front != 0)

{

rear = 0;

queue[rear] = value;

}

else {

rear++;

queue[rear] = value;

}

printf("Inserted %d\n", value);

}

void del() {

if (front == -1) {

printf("Queue is Empty\n");

return;

}

printf("Deleted %d\n", queue[front]);

queue[front] = -1;

if (front == rear) {

front = rear = -1;

}

else if (front == SIZE - 1) {

front = 0;

}

else {

front++;

}

}

void display() {

if (front == -1) {

printf("Queue is Empty\n");

return;

}

printf("Queue elements are: ");

if (rear >= front) {

for (int i = front; i <= rear; i++)

printf("%d ", queue[i]);

}

else {

for (int i = front; i < SIZE; i++)

printf("%d ", queue[i]);

for (int i = 0; i <= rear; i++)

printf("%d ", queue[i]);

}

printf("\n");

}

int main() {

int choice, value;

while (1) {

printf("\n1. Insert\n2. Delete\n3. Display\n4. Exit\n");

printf("Enter your choice: ");

scanf("%d", &choice);

switch (choice) {

case 1:

printf("Enter the value to insert: ");

scanf("%d", &value);

insert(value);

break;

case 2:

del();

break;

case 3:

display();

break;

case 4:

exit(0);

default:

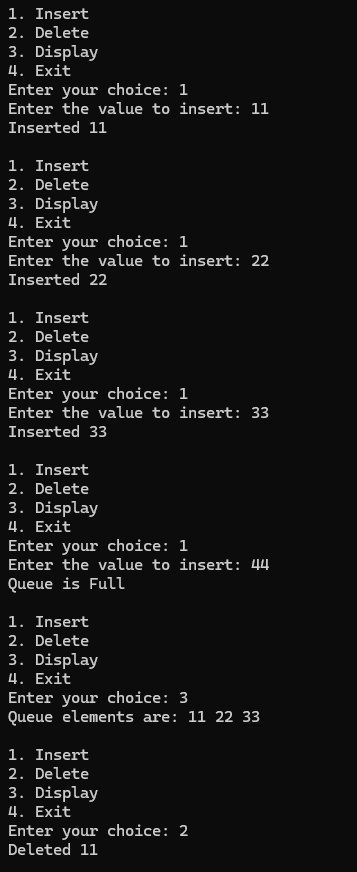
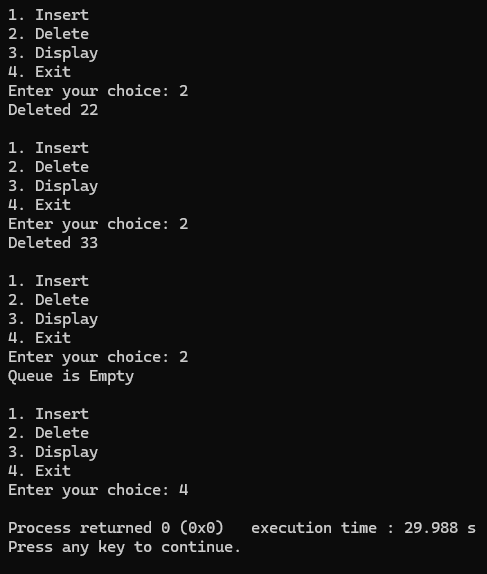
printf("Invalid choice\n");

}

}

return 0;

}

**Leetcode**

Give a string s, find the first non repeating character in it and return its index. If it does not exist, return -1.

int firstUniqChar(char\* s) {

    int flag = 0 , i , j;

    for( i = 0 ; s[i] ; i++ ) {

        for( j = 0 ; s[j] ; j++ , flag = 0 ) {

            if( ( s[i] == s[j] ) && ( i != j ) ) {

                flag = 1;

                break;

            }

        }

        if(flag == 0)

        return i;

    }

    return -1;

}

