Lab 3b:CIRCULAR QUEUE

SHREYA S RUDAGI
1BM22CS267

```
#include <stdio.h>
#define N 5
int q[N];
int front = -1, rear = -1;
void insert(int);
int delete();
void display();
void main()
{
   int n, choice;
   do
   {
      printf("\n1.Insert\n2.Delete\n3.Display\n4.Exit\n");
      printf("Enter your option : \n");
      scanf("%d", &choice);
      switch (choice)
      {
      case 1:
         printf("Enter the number to be inserted in the queue : \n");
         scanf("%d", &n);
         insert(n);
         break;
      case 2:
         n = delete ();
         if (n != -1)
            printf("\n The number deleted is : %d\n", n);
         break;
      case 3:
         display();
         break;
      case 4:
         exit(0);
         break;
      default:
         printf("Invalid option\n");
         exit(0);
         break;
      }
   } while (choice != 4);
}
void insert(int num)
{
   if ((front == 0 && rear == N - 1) || (rear == (front - 1)))
      printf("\n OVERFLOW");
   else if (front == -1 && rear == -1)
   {
      front = rear = 0;
      q[rear] = num;
```

```
    }
    else if (rear == N - 1 && front != 0)
    {
        rear = 0;
        q[rear] = num;
    }
    else
    {
        rear++;
        q[rear] = num;
    }
}
int delete()
{
    int val;
    if (front == -1 && rear == -1)
    {
        printf("\n UNDERFLOW");
        return -1;
    }
    val = q[front];
    if (front == rear)
        front = rear = -1;
    else
    {
        if (front == N - 1)
            front = 0;
        else
            front++;
    }
    return val;
}
void display()
{
    int i;
    printf("\n");
    if (front == -1 && rear == -1)
        printf("\n QUEUE IS EMPTY");
    else
    {
        if (front < rear)
        {
            for (i = front; i <= rear; i++)
                printf("\t %d", q[i]);
        }
        else
        {
            for (i = front; i < N; i++)
                printf("\t %d", q[i]);
            for (i = 0; i <= rear; i++)
```

```
        printf("\t %d", q[i]);
  }
```

```
1.Insert
2.Delete
3.Display
4.Exit
Enter your option :1
Enter the number to be inserted in the queue : 1

1.Insert
2.Delete
3.Display
4.Exit
Enter your option :1
Enter the number to be inserted in the queue : 2

1.Insert
2.Delete
3.Display
4.Exit
Enter your option :1
Enter the number to be inserted in the queue : 3

1.Insert
2.Delete
3.Display
4.Exit
Enter your option :1
Enter the number to be inserted in the queue : 3

1.Insert
2.Delete
3.Display
4.Exit
Enter your option :3

         1       2       3       3
1.Insert
2.Delete
3.Display
4.Exit
Enter your option :
4

Process returned 0 (0x0)   execution time : 20.489 s
Press any key to continue.
```