```c
#include<stdio.h>
#include<conio.h>
#define MAX 10

int q[MAX];
int front=-1;
int rear=-1; // global variables, so that any function can access these

void insert_f(int n) // insertion at the front
{
        int i;

        if(front==-1) // in case of 1st entry to the queue
        {
                front++;
                rear++;
                q[front] = n;
        }

        else if(front==0 && rear==MAX-1)
        {
                printf("Overflow!");
                exit(0);
        }
        else if(front==0 && rear!=MAX-1)
        {
                for(i=rear; i>=front; i--)
                        q[i+1]=q[i]; // right shift of all elements
                q[front] = n;
                rear++;
        }
        else
        {
                front--;
                q[front] = n;
        }
}

void insert_r(int n) // insertion at the rear
{
        int i;

        if(front==-1) // in case of 1st entry to the queue
        {
                front++;
                rear++;
                q[front] = n;
        }

        else if(front==0 && rear==MAX-1)
        {
                printf("Overflow!");
                exit(1);
        }
        else if(front!=0 && rear==MAX-1)
```

```c
        {
                for(i=front; i<=rear; i++)
                        q[i-1]=q[i]; // left shift of all elements
                q[rear] = n;
                front--;
        }
        else
        {
                rear++;
                q[rear] = n;
        }
}

int del_f() // deletion from the front
{
        int n;
        if(front==-1)
        {
                printf("Underflow!");
                exit(2);
        }
        n = q[front];
        if(front==rear) // when the queue has a single element
        {
                front=-1;
                rear=-1;
        }
        else
                front++;
        return n;
}

int del_r() // deletion from the rear
{
        int n;
        if(front==-1)
        {
                printf("Underflow!");
                exit(3);
        }
        n = q[rear];
        if(front==rear) // when the queue has a single element
        {
                front=-1;
                rear=-1;
        }
        else
                rear--;
        return n;
}

void main()
{
        int item,i;
        clrscr();
```

```c
        insert_r(10);
        insert_r(20);
        insert_f(30);
        insert_f(40);

        item = del_f();
        printf("Deleted item is.. %d\n",item);

        item = del_r();
        printf("Deleted item is.. %d\n",item);

        getch();
}
```