NAME:SWETHA.J

ROLLNO:230701357

EX-7: Implementation of Queue using Array and Linked list

LINKED LIST IMPLEMENTATION-QUEUE

```
#include <stdio.h>
#include<stdlib.h>

struct node
{
    int data;
    struct node *link;
}*F=NULL,*R=NULL;

int IsEmpty();
void Enqueue(int);
void Dqueue();
void Display();
int IsEmpty()
{
    if(F==NULL&&R==NULL)
    {
        return 1;
    }
    else
    return 0;
}
void Enqueue(int val)
{
    struct node*newnode;
    newnode=(struct node*)malloc(sizeof(struct node));
    newnode->data=val;
    if(IsEmpty()){
        F=R=newnode;
    }
    else
    {
        R->link=newnode;
        R=newnode;
    }
    newnode->link=NULL;
}
void Dqueue()
{
    struct node*temp=F;
    if(IsEmpty())
    {
        printf("list is empty");
    }
    else
    {
        printf("\nDeleted element is: %d",temp->data);
        if (F==R)
        F=R=NULL;
        else
        F=F->link;
        free(temp);

    }
}
void Display()
{
    struct node*temp=F;
    if(IsEmpty())
    {
        printf("underflow");
    }
    else
    {
        while(temp!=NULL)
        {
            printf("\n%d",temp->data);
            temp=temp->link;
        }
    }
}
int main()
{
    int choice,t=1,val;
    while (t==1)
    {
        printf("\n\n\nMENU\n");
        printf("1.Insert an element\n2.Delete an element\n3.Display the 
Queue\n4.EXIT\n");
        printf("\nEnter your choice:");
        scanf("%d",&choice);
        switch (choice)
        {
            case 1:
            printf("Enter the value to be inserted:");
            scanf("%d",&val);
            Enqueue(val);
            break;
            case 2:
            Dqueue();
            break;
            case 3:
            Display();
            break;
            case 4:
            t=0;

        }
    }
}
```

ARRAY IMPLEMENTATION-QUEUE

```
#include<stdio.h>
#include<stdlib.h>
#define size 5
```

```
int que[size];
void Enqueue(int);
void Dqueue();
void Display();
int IsFull();
int IsEmpty();
int F=-1,R=-1;
int IsFull()
{
    if (size-1==R)
    {
        return 1;
    }
    else
    return 0;
}
int IsEmpty()
{
    if(F==-1)
    return 1;
    else
    return 0;
}
void Enqueue(int data)
{
    if(IsFull())
    {
        printf("overflow");
    }
    else if(F==-1)
    {
        F=0;
    }
    R=R+1;
    que[R]=data;
}

void Dqueue()
{
    if(IsEmpty())
    {
        printf("underflow");
    }
    else
    {
       printf("Deleted Element is:%d",que[F]);
       if (R==F)
          R=F=-1;
       else
          F=F+1;
    }
}
void Display()
{
    if(IsEmpty())
    {
        printf("No elements in queue");
    }
    else
    {
        for(int i=F;i<=R;i++)
        {
            printf("%d\n",que[i]);
        }
    }
}
int main()
{
    int choice,t=1,val;
    while (t==1)
    {
        printf("\n\n\nMENU\n");
        printf("1.Insert an element\n2.Delete an element\n3.Display the
Queue\n4.EXIT\n");
        printf("\nEnter your choice:");
        scanf("%d",&choice);
        switch (choice)
        {
            case 1:
            printf("Enter the value to be inserted:");
            scanf("%d",&val);
            Enqueue(val);
            break;
            case 2:
            Dqueue();
            break;
            case 3:
            Display();
            break;
            case 4:
            t=0;

        }
    }
}
```

```
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT

Enter your choice : 1

Enter the element : 10

1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT

Enter your choice : 1

Enter the element : 20

1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT

Enter your choice : 1

Enter the element : 30

1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
```

```
Enter your choice : 1
Enter the element : 40
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 1
Enter the element : 50
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 1
Enter the element : 60
Queue is Overflow...!
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 3
10 20 30 40 50
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 2
10
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 2
20
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 2
30
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 2
40
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 2
50
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
Enter your choice : 2
Queue is Underflow...!
1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT
```

```
Enter your choice : 3

Queue Underflow...!

1.ENQUEUE 2.DEQUEUE 3.DISPLAY 4.EXIT

Enter your choice : 4
```