LAB PROGRAM—4 (PRACTICE)

4) Simulate the working of a double ended queue which is input restricted and output restricted.

Code:
```
#include<stdio.h>
#define qsize 5
int f=0,r=-1,ch;
int item,q[10];
int isfull()
 {
 return(r==qsize-1)?1:0
 }
 int isempty()
  {
  return(f>r)?1:0;
 }
void insert_rear()
  {
  if(isfull())
  {
   printf("queue is empty\n");
   return;
   }
 printf("item deleted is %d\n",q[(f)++]);
  if(f>r)
   {
   f=0
   r=-1;
   }
  }
```

PAGE No.

DATE / /

```
void insert-front()
  {
   if (f!=0)
    {
     f=f-1;
    q[f]=item;
    return;
    }
  else
   printf("insertion is not possible\n");
  }
void delete_rear()
  {
  if(isempty())
    {
     printf("queue isempty \n");
    return;
    }
printf("item deleted is %d\n", q[(r)--]);
 if(f>r)
  {
  f=0;
  r=-1;
 }}
void display()
 { int i;
  if(isempty())
  {printf("queue empty\n");
   return;
  }
for(i=f;i<=r;i++)
      printf("%d\n",q[i]);
 }
```

Scanned with CamScanner

```
void main()
 {
 for(;;)
  {
  printf("1.insert_rear\n 2.insert_front\n 3.del
rear\n 4.delete_front \n 5-display 6.exit\n");
 printf("Enter choice \n");
  scanf("%d", &ch);
 switch(ch)
   {
   case 1: printf("enter the item\n");
            scanf("%d", &item);
           insert_rear();
            break;
  case 2: printf("enter the item \n");
              scanf("%d", &item);
             incert_front();
            break;
  case 3: delete_rear();
           break;
   case 4; delete_front();
           break;
   case 5: display();
           break;
  default: exit(0);
 }
}
}
```

Q WAP to simulate input restricted deque.

Ans
```
#include <stdio.h>
#define qsize 5
int f=0, r=-1, ch;
int item, q[10];
 int isfull()
   {
    return(r==qsize-1)?1:0;
   }
 int isempty()
  {
  return(f>r)?1:0;
  }
 void insert_rear()
  {
  if(isfull())
    {
   printf("queue overflow\n");
   return;
   }
  r=r+1;
   q[r]=item;
 }
void delete_front()
 {
 if(isempty())
 {
 printf("queue empty\n");
  return;
  }
printf("item deleted is %d\n", q[(f)++]);
```

```
if(f>r)
 {
 f=0;
  r=-1;
 }}
void delete_rear()
  {
  if (isempty())
   {
   printf("queue is empty\n");
   return;
  }
printf("item deleted is %d\n";q[(r)--]);
if(f>r)
 {
 f=0;
  r=-1;
 }
}
void display()
 { int i;
if(isempty())
 {
 printf("queue empty\n");
 return;
 }
for(int i=f;i<=r;i++)
  printf("%d\n",q[i]);
 }
void main()
 {
 for(;;)
```

```
{
printf("1. inset. rear \n2. delete-rear \n3. delete-
  front \n4. display \n5. exit");
  printf("Enter choice");
  scanf("%d", &ch);
  switch(ch)
  {
  case 1: printf("enter the item\n");
            scanf("%d", &item);
            insert_rear();
            break;
 case 2: ~~p:~~ delete_rear();
            break;
 case 3: delete_front();
            break;
case 4: display();
            break;
 default: exit(0);
  }}
  }
```

Q. WAP to simulate output restricted deque

ans
```
#include <stdio.h>
#define qsize 5
int f=0,r=-1,ch;
int item,q[10];
int isfull()
{
 return(r==qsize-1)?1:0;
 }
 int isempty()
   {
   return(f>r)?1:0;
   }
void insert_rear()
{
 if(isfull())
   {
   printf("queue overflow\n");
   return;
  }
r=r+1;
q[r]=item;
 }
void delete_front()
 {
  if(isempty())
    {
     printf("queue empty\n");
     return;
    }
 printf("item deleted is %d\n",q[(f)++]);
   if(f>r)
{
f=0;
r=-1;
}}
void insert_front()
{
 if(f!=0)
  { ~~f=~~ f=f-1;
  q[f]=item;
  return;
  }
else if((f==0)&&(r==-1))
  {
  q[++(r)]=item;
   return;
 }
else
  printf("insertion not possible\n");
 }
void display()
{
int i;
 if (isempty())
  {
   printf("queue empty\n");
  return;
  }
for(i=f;i<=r;i++)
printf("%d\n",q[i]);
 }
void main()
```

```
{
for(;;)
 {
 printf("1. insert_rear\n 2. insert_front\n
   3. delete_front \n4. display \n5. exit\n");
printf("enter choice\n");
 scanf("%d", &ch);
 switch(ch)
   {
case 1: printf("enter the item\n");
          scanf("%d", &item);
          insert_rear();
          break;
case 2: printf("enter the item\n");
          scanf("%d", &item);
          insert_front();
          break;
case 3: delete_front();
          break;
case 4: display();
          break;
default: exit(0);
 }
 }
}
```