Lab:

Linear Queue

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
int item, front=0, rear=-1, q[10];
void insertrear()
{
if (count == que.SIZE-1)
{
printf ("queue overflow");
return;
}
rear = (rear+1);
q[rear] = item;
count++;
}
int deletefront ()
{
if (front > rear)
{
front=0;
rear = -1;
return -1;
}
return q[front++];
}
void displayq ()
{
int i;
if (front > rear)
```

```
{
printf ("queue is empty\n");
return;
}
printf ("contents of queue\n");
for (i=front ; i<=rear ; i++)
{
printf ("%d\n", q[i]);
}
}
int main()
{
int choice;
for(;;)
{
printf ("1: insertrear 2: deletefront 3: display
        4: exit\n");
printf ("enter the choice\n");
scanf ("%d", &choice);
switch (choice)
{
case 1: printf ("enter the item to be inserted\n");
        scanf ("%d", &item);
        insertrear();
        break;
case 2: item = deletefront();
        if (item == -1)
```

```
printf("queue is empty\n");
else
printf("item deleted = %d\n", item);
break;
case 3: display();
        break;
        default: exit(0);
    }
  }
}
```

Output:

1: insert 2: delete from front 3: display 4: exit

enter the choice

1

enter the item to be inserted

10

enter the choice

3

contents of queue

10.

```
1:insertrear 2:deletefront 3:display 4:exit
enter the choice
1
enter the item to be inserted
10
1:insertrear 2:deletefront 3:display 4:exit
enter the choice
1
enter the item to be inserted
20
1:insertrear 2:deletefront 3:display 4:exit
enter the choice
3
contents of queue
10
20
1:insertrear 2:deletefront 3:display 4:exit
enter the choice
2
item deleted=10
1:insertrear 2:deletefront 3:display 4:exit
enter the choice
10
```

Circular Queue

Page No.
Date 19/10/2020

```
#include <stdio.h>
#include <stdlib.h>
#define que_SIZE 3
int item, front=0, rear=-1, q[queSIZE], count=0;
void insertrear()
{
if(count==que_SIZE)
{
printf("queue overflow");
return;
}
rear=(rear+1)%que_SIZE;
q[rear]=item;
count++;
}
int deletefront()
{
if(count==0) return -1;
item=q[front];
front=(front+1)%que_SIZE;
count=count-1;
return item;
}
void display()
{
int i,f;
if(count==0)
{
```

```
printf ("queue is empty");
return;
}
f = front;
printf ("contents of queue \n");
for (i=0; i<= count; i++)
{
printf ("%d \n", q[f]);
f = (f+1) % que_size;
}
}
Void main()
{
int choice;
for (;;)
{
printf ("\n1. Insert rear \n 2. Delete front \n
        3. Display \n4. exit \n");
printf (" enter the choice");
Scanf ("%d \n", &choice);
switch (choice)
{
case 1: printf ("Enter the item to be inserted");
        scanf ("%d", &item);
        insertrear();
        break;
case 2: item = deletefront();
        if (item == -1)
        printf ("queue is empty \n");
```

```
else
printf("item deleted is %d\n", item);
break;
case 3: display();
        break;
default: exit(0);
    }
  }
```

output:

1. insertrear
2. deletefront
3. display
4. exit

Enter the choice : 1
Enter the element to be inserted : 10
Enter the choice : 1
Enter the element to be inserted : 20
Enter the choice : 3
Contents of queue
10
20

```
1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 1
Enter the item to be inserted :10

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 2
item deleted is 10

1.Insert rear
2.Delete front
3.Display
4.exit
 Enter the choice : 1
Enter the item to be inserted :20

1.Insert rear
2.Delete front
3.Display
```