19/10/20

LAB-PROGRAM : 3

NAME : SAIPRAVEEN MARNI
USN : 1BM19CS138

CIRCULAR QUEUE

```
#include<stdio.h>
#include<stdlib.h>
#include<process.h>
#define que_size 3
int item, front = 0, rear = -1, q[que_size], count = 0;
void insertreal()
{
 if (count == que_size)
 {
  printf("queue overflow");
    return;
 }
 real = (real+1) %. que size;
    q[real] = item;
    count ++;
}
 int  delete front()
 {
   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 real \n2: Delete front \n3: Display \n4: exit\n");
  printf ("Enter the choice:");
  scanf ("%d", &choice);
  switch (choice)
{
case 1: printf ("Enter item to be inserted:");
        scanf ("%d", &item);
        insertreal();
        break;
case 2: item = deletefront();
  if (item == -1)
printf ("queue is empty \n");
  else
printf ("item deleted is %d \n", item);
    break;
Case 3: displayq();
```

```
        break();
    default: exit(0);
    }
}
```

LINEAR QUEUE

```
#include <stdio.h>
#include <stdlib.h>
#define QUE_SIZE 3
int item, front = 0, rear = -1, q[10];
  void insertrear()
{ if (rear == QUE_SIZE - 1)
  {
    printf("queue overflow \n");
      return;
  }
    rear = rear + 1;
    q[rear] = item;
}
  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: insert rear\n2: delete front\n3: display\n4: 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);
                insert rear();
                break;
        Case 2: item = delete front();
                if(item == -1)
                printf("queue is empty\n");
                else
                printf("item deleted = %d\n", item);
                break;
        Case 3: display Q()
                break;
                default: exit(0);
        }
    }
}
```