SHIVANI GAHLOT
1BM19CS150

LAB-3

WAP to simulate the working of a queue of integers using an array. Provide the following operations
(a) Insert
(b) Delete
(c) Display
The program should print appropriate messages for queues empty and queue overflow conditions

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 displayQs()
{ 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\n 2: deletefront\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);
    insertrear();
    break;
    case 2: item = deletefront();
    if(item == -1)
    printf("item deleted=%d\n", item);
    break;
    case 3: display();
    break;
    default: exit(0);
}
}
}
```

CIRCULAR QUEUE

```
#include <stdio.h>
#include <stdlib.h>
#define que_size 3
int item, front=0, rear=-1, q[que_size], 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 displayq()
{
  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\n2:Delete front\n3:Display\n4:exit\n");
   printf("Enter the choice; ");
   scanf("%d", &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: displayq();
             break;
     default : exit(0);
   }
  }
}
```