Practicle No-6

6.Write a to Make a Queue And Insert ,delete Element's.

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
struct queue{
    int size;
    int f;
    int r;
    int*arr;

};
int isEmpty(struct queue*q){
    if(q->r==q->f){
        return 1;
    }
    return 0;
}
int isFull(struct queue*q){
    if(q->r==q->size-1){
        return 1;
    }
    return 0;
}
void enqueue(struct queue*q,int val){
    if(isFull(q)){
        printf("This Queue is Full");
    }
    else{
        q->r++;
        q->arr[q->r]=val;
    }
}
int dequeue(struct queue*q){
    int a=-1;
    if(isEmpty(q)){
        printf("Queue is Empty");
    }
    else{
        q->f++;
        a=q->arr[q->f];
    }
    return a;
}
int main()
{
    int size_ofQueue;
    printf("Enter Size of Queue:\n");
    scanf("%d",&size_ofQueue);
struct queue q;
q.size=size_ofQueue;
q.f=q.r=-1;
```

```
q.arr=(int*)malloc(q.size*sizeof(int));
int p,n,i,m;
printf("Enter a Number of Element's:\n");
scanf("%d",&n);
printf("Enter Element's for Enqueuing:\n");
for(i=0;i<n-1;i++){
    scanf("%d\n",&m);
    enqueue(&q,m);
}
printf("Dequeuing is %d\n",dequeue(&q));
printf("Dequeuing is %d\n",dequeue(&q));
printf("Dequeuing is %d\n",dequeue(&q));
if(isEmpty(&q)){
    printf("Queue is Empty\n ");
}
return 0;
}
//code is Runig with output
```