WAP do simulate the working of a queue of intergey using an array, Provide the following operations. a) insert b) Delete c) Display. The program should print appropriate message for queue empty & overflow:

```
#include <stdio.h>
# include <conio.h>
#include <stdlib.h>
#define QSIZE 5
int item, front = 0, rear = -1, q[10];
void insertrear(){
   if (rear == QSIZE-1){
     printf("Queue is OverFlow");
      return;
   } rear += 1;
    q[rear] = item; }
int deletfront(){
   if (front > rear){
      front = 0;
      rear = -1;
        return -1;
   } return q[front++];
 }
 void display (){
   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]);
 }
```

low:

```
void main()
{ char ch;
   int choice;
   for( ; ; ){
printf("1-insert rear \n 2. delete front \n 3.
      display \n 4. exit : \n");
 printf("enter choice \n");
  scanf("%d", &ch);
 switch (ch)
  {
    case 1: printf("enter the item : \n");
            scanf("%d", &item);
              insertrear();
            break;
      case 2 : item = deletfront();
       if (item == -1){
          printf("Queue is Underflow\n");
               break; }
         printf("item Deleted : %d \n", item);
         break;
       Case 3 : display();
        break;
      default : exit (0);
    }
  }
}
```

#include<stdio

Circular Queue:

1) Insert
2) Delete
3) Display.

```
#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 insertrear() {
    if(count == que-size)
    { printf("queue overflow\n");
      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 queu \n");
for (i=0 ; i<=count ; i++){
  f(f+1)%que_size;
   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. Exist\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 ("itemdeleted is %d \n", item);
           break;
       case 3 : displayq();
          break;
     default : exit(0);
  }
 }
}
```