Circular Queue

```
#include <stdio.h>
# define q-size 5 .
int item, front =0, rear =-1 ,q[q-size], count=0;
void insert_rear ()
{ if( count ==q-size)
  {
   printf ("Queue Over flow \n");
   return ;
  }
 rear =(rear+1)%q-size ;
 q [rear] =item;
 count ++;
 return;
}
int delete-front ( )
{ if (count==0)
   return -1;
  else{
   item =q[front];
   front=(front+1) %q-size;
   count =count -1;
   return item ;
  }
```

```
}
void display()
 {
   int i,f;
   if(counter==0)
    {
     printf("queue is empty\n");
     return;
    }
   f=front;
   printf("contents of Queue:\n");
   for(i=0;i<=count;i++)
    {
      printf("%d\n",q[f]);
      f=(f+1)%q-size;
 }
}
void main()
{
 int choice;
 for(;;)
 {
  printf("1:Insert \n 2:DELETE \n 3:DISPLAY")
  printf("enter the choice \n");
  scanf("%d", &choice);
 switch(choice) {
```

```
case 1 : printf ("Enter an element to be Inserted \n");
         Scanf (" %d", &~~choice~~);
             insert_read ();
             break;
case 2 : item=deleted-front. ();
         if (item ==-1)
            printf ( "queue is empty \n");
           printf (" item deleted is %d \n", item);
         break;
case 3 : display();
           break;
default : exit (0) ;
  }

 }
};
```

Linear Queue

```
#include <stdio.h>
~~#~~ #define MAX 5.
 int queue[MAX];
int rear = -1 , front = -1;
void insert()
 { int add_item;
  if ( rear == MAX -1 )
  printf ("Queue Overflow in");
  else
   { if (front == -1 )
             front == 0;
   printf (" enter the element to be inserted : ');
   scanf ("%d", &add_item);
   rear = rear + 1;
   queue[rear] = add_item;
  }
 }
void delete()
{
 if (front == -1 || front > rear )
  { printf (" Queue underflow in");
  return;
  }
```

```
else
  {printf ("Element deleted from queue is : %\n",
              queue [front]);
   front = front + 1;
  }

void display ()
{ int  i;
  if  (front == -1 )
   printf ("queue is empty \n");
  else
   { printf ( "queue is : \n");
   ~~from~~ for (i= front ; i <= rear ; i++ )
              printf("%d\n", queue [i] );
          print ("\n");
     }
    }

    main ()
    {
     int choice ;
     while (1)
    {
      printf ("1: INSERT \n");
      printf ("2: DELETE \n");
     printf
          ("3 : DISPLAY \n");
     printf (" 4 : EXIT\n");
printf ("Enter your choice :");
`scanf ("%d", &choice );
switch (choice ) :
{
  case 1 :
   insert();
   break;
 case 2 :
 delete ();
 break ;
  case 3:
 display();
  break;
 case 4:
 exit ( 1);
 default :
printf("wrong choice\n' );
}
}
}
```

```
printf ("Enter your choice : ");
scanf ("%d", &choice );
switch (choice ) :
{
  case 1 :
   insert();
   break;
  case 2 :
  delete ();
  break ;
   case 3:
  display();
   break;
  case 4:
  exit ( 1);
  default :
printf ("wrong choice\n' );
}
}
}
```