Circular queue.

```c
# 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  insert rear()
{
    if ( count == que size)
    {
        printf (" Queue ouerflow");
        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 ("\n 1. Insert rear \n 2. Delete from
                 \n 3. display \n 4. Exit \n");
        printf ("Enter the choice: ");
        scanf ("%d", &choice);
        switch (choice)
        {
            case 1: printf ("Enter the item to
                             be inserted: ");
                    scanf ("%d", &item);
                    insert rear ();
                    break;
            case 2: item = delete front ();
                    if (item == -1)
                        printf ("Queue is empty \n");
                    else
                        printf ("Item deleted is: %d");
```

```
                break ;
    case 3:  display9();
                    break ;
    default :  exit(0);
    }
}
}
```

Queue

```c
#include <stdio.h>
#include <stdlib.h>
#define Que_size 3.
int item, front=0, rear=-1, q[0];
void insert rear ()
{ 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 display Q()
{ 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]);
} }
```

```c
int main ()
{
    int choice;
    for (;;)
    {
        printf ("1) Insert rear\n 2) Delete front
                \n 3) Display \n 4) Exit\n");
        printf ("Enter your choice:");
        scanf ("%d", &choice);
        switch (choice)
        {
            case 1: print ("Enter the item
                            to be inserted\n");
            scanf ("%d", &item);
            insert rear ();
            break;
            case 2: item = deletefront ();
            if (item == -1)
            printf (" Queue is empty\n");
            else
            printf (" Item deleted = %d\n",
                                    item);
            break;
            case 3: display q ();
            break;
            default : exit (0);
        }
    }
}
```