

```

#include "cqueue.h"
cqueue makenullcqueue(int sz)
{
    cqueue t;
    t=(cqueue) malloc(sizeof(struct arrcqueue));
    t->a=(int *) malloc(sizeof(int)*sz);
    t->m=sz;
    t->n=0;
    t->f=0;
    t->r=0;
    return t;
}
void printoptions()
{
    printf("\n0.Exit");
    printf("\n1.Enqueue");
    printf("\n2.DeQueue");
}
int getoption()
{
    int opt;
    printf("\nEnter an Option:");
    scanf("%d",&opt);
    return opt;
}
position frontpos(cqueue cq)
{
    return cq->f;
}
position rearpos(cqueue cq)
{
    return cq->r;
}
position nextpos(cqueue cq,position p)
{
    return p+1;
}
int isempty(cqueue cq)
{

```

```

        if (cq->n==0)
            return 1;
        return 0;
    }
int isfull (cqueue cq)
{
    if (cq->n==cq->m-1)
        return 1;
    return 0;
}
void enqueue (cqueue cq, element e)
{
    cq->a[cq->r]=e;
    cq->r=(cq->r+1)%(cq->m);
    cq->n++;
}
element dequeue (cqueue cq)
{
    int i;
    element e;
    e=cq->a[cq->f];
    cq->f=(cq->f+1)%(cq->m);
    cq->n--;
    return e;
}
void printqueue (cqueue cq)
{
    position i;
    printf("\nElements in the queue are:\n");

    for (i=frontpos (cq); i!=rearpos (cq); i=(nextpos (cq,
i))%(cq->m))
        printf ("%d ", cq->a[i]);
}

```