```c
#include"queue.h"
queue makenullqueue(int sz)
{
    queue t;
    t=(queue)malloc(sizeof(struct arrqueue));
    t->a=(int *)malloc(sizeof(int)*sz);
    t->m=sz;
    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(queue q)
{
    return q->f;
}
position rearpos(queue q)
{
    return q->r;
}
position nextpos(queue q,position p)
{
    return p+1;
}
int isempty(queue q)
{
    if(q->r==0)
```

```
        return 1;
    return 0;
}
int isfull(queue q)
{
    if(q->r==q->m)
        return 1;
    return 0;
}
void enqueue(queue q,element e)
{
    q->a[q->r]=e;
    q->r++;

}
element dequeue(queue q)
{
    int i;
    element e;
    e=q->a[q->f];

for(i=frontpos(q);i<rearpos(q);i=nextpos(q,i))
        q->a[i]=q->a[i+1];
    q->r--;
    return e;
}
void printqueue(queue q)
{
    position i;
    printf("\nElements in the queue are:\n");

for(i=frontpos(q);i<rearpos(q);i=nextpos(q,i))
        printf("%d ",q->a[i]);
}
```