```c
#include"queuell.h"
queuell makenullqueuelist()
{
    queuell t;
    t=(queuell)malloc(sizeof(struct node));
    t->next=NULL;
    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(queuell q)
{
    return q;
}
position rearpos(queuell q)
{
    position i=q;
    while(i->next!=NULL)
        i=i->next;
    return i;
}
position nextpos(queuell q,position p)
{
    return p->next;
}
int isempty(queuell q)
{
    if(q->next==NULL)
```

```
        return 1;
    return 0;
}
void enqueue(queuell q,element e)
{
    position p=rearpos(q);
    queuell t;
    t=makenullqueuelist();
    t->data=e;
    t->next=p->next;
    p->next=t;
}
element dequeue(queuell q)
{
    position i=frontpos(q)->next;
    element e;
    e=i->data;
    q->next=i->next;
    free(i);
    return e;
}
void printqueue(queuell q)
{
    position i;
    printf("\nElements in the queue are:\n");

for(i=frontpos(q);i!=rearpos(q);i=nextpos(q,i))
        printf("%d ",i->next->data);
}
```