Pro09-8 write a program to implement queue cesing linked list? #include < st dio. n> It include Lonio. hs # include 2 std lib. ho Stouct node ent is E ina info; Struct node * line; 3 xf=NULL, * 7=NULLi in mains € in+() Clascacoi while (1) ¿ prinap (" In MAIN MENU"); Print ("In linsert"); Point-P (" In 2- delete"); Point & (aln 3. display"); Print ("In 4. Quit"); Printf C"In Enter the Choice") scanf ("In -1.d", & c)i

```
Switch (e)
 Case 1: Push ();
             breaki
   Casez: POPCD;
             break;
   cases: displayeri
               palak.
   Case 4: exit(1);
              bolak;
  default: Prints ("wrong");
   Pushes
  2 Strouget a nocle * t;
     int item;
 t = (Street node +) malloc (Size of (Struct node))
 Prince (" In insert the dement"),
  Scanf (" 1.d") gitemi
   t-info = ifem;
   E- IIAK = NULL i
```

```
if (f == NULL)
  feti
 else
(V-) link . ti
 DEti
 & (funn;
3 porcs
 E Struct node & +;
   if (f= NULL)
   points ( " In gracue is empty"),
   Pise
   { t=f;
Printf ["In delete the I.d element", (-) info).
   f= f -> link)
    hee(t);
  3 geturn:
    3 dis Play ()
E Struct noch & Droi
     pto sfi
   if (F= NUCL)
 Paint & " In que ce is empty m),
```

eye Epoint (" que a element ase ! In"); enhile (Pto 1 = NULL) E point ("Yed In") Pla -) into) 1 Pars Ptr-> link; prings ("(n").

OUTPUT

MAINMENU

1. Insert

2-duete

3. display

4. Cort

enses In choice 1

ingest the element: 3

MAIN MENU

1 insent

2. de lete

3. display

4. exit

enser su choice: 3.

que us clement a se:

```
NTURBOC3NDHANYANQUEUELIN.C —
#include<stdio.h>
#include<comio.h>
#include<stdlib.h>
struct node
int info:
struct node *link:
*f=NULL,*r=NULL;
int main()
int c:
clrscr();
while(1)
printf("\nMAIN MENU");
printf("\n1.Insert");
printf("\n2.delete");
printf("\n3.display");
printf("\n1.Quit");
printf("\n enter the choice");
       82:7
```

```
\TURBOC3\DHANYA\QUEUELIN.C ---
printf("\n enter the choice");
scanf ("zd", &c);
switch(c)
case 1: push();
         break:
case 2: pop():
         break;
case 3: display();
        break:
case 4: exit(1);
        break:
default: printf("wrong");
push()
struct node* t:
int item:
t=(struct node*)malloc(sizeof(struct node));
       34:52 ---
```

```
NTURBOC3NDHANYANQUEUELIN.C ——
                                                                         -1=[‡]=n
t=(struct node*)malloc(sizeof(struct node));
printf("\n insert the element");
scanf ("zd",&item);
t->info=item:
t->link=NULL:
if(f==NULL)
f=t:
else
r->link=t:
r=t:
return:
pop()
struct node *t:
if (f ==NULL)
printf("\nqueue empty");
else
t=f:
printf("\ndelete the %d element ",t->info);
     - 61:52 -----
```

```
-[ 🛮 ] ---
                          \TURBOC3\DHANYA\QUEUELIN.C =
printf("\ndelete the %d element ",t->info);
f=f->link:
free(t);
return;
display()
struct node *ptr;
ptr=f;
if (f ==NULL)
printf("\n queue is empty");
else
printf("queue element are:\n");
while(ptr!=NULL)
printf("xd\n",ptr->info);
ptr=ptr->link;
       70:54 ---
```

```
-[ 🛮 ] ---
                          \TURBOC3\DHANYA\QUEUELIN.C =
printf("\ndelete the %d element ",t->info);
f=f->link:
free(t);
return;
display()
struct node *ptr;
ptr=f;
if (f ==NULL)
printf("\n queue is empty");
else
printf("queue element are:\n");
while(ptr!=NULL)
printf("xd\n",ptr->info);
ptr=ptr->link;
       70:54 ---
```

```
\TURBOC3\DHANYA\QUEUELIN.C —
display()
struct node *ptr:
ptr=f:
if (f ==NULL)
printf("\n queue is empty");
else
printf("queue element are:\n");
while(ptr!=NULL)
printf("xd\n",ptr->info);
ptr=ptr->link;
printf("\n");
return;
       86:54 ---
```

```
MAIN MENU
1. Insert
2.delete
3.display
4.Quit
 enter the choice 1
 insert the element 3
MAIN MENU
1. Insert
2.delete
3.display
4.Quit
 enter the choice 3
queue element are:
3
MAIN MENU
1. Insert
2.delete
3.display
4.Quit
 enter the choice
```