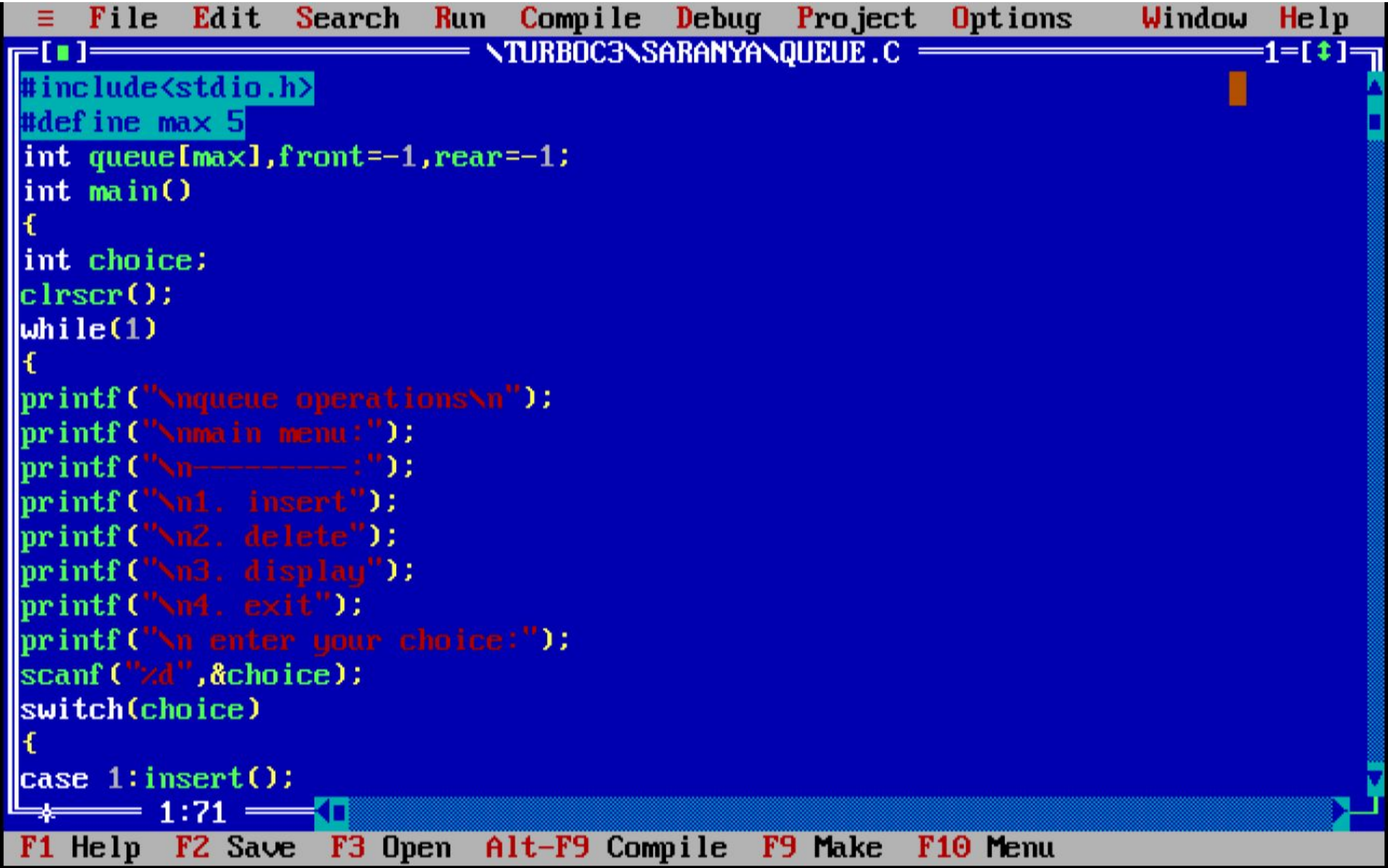
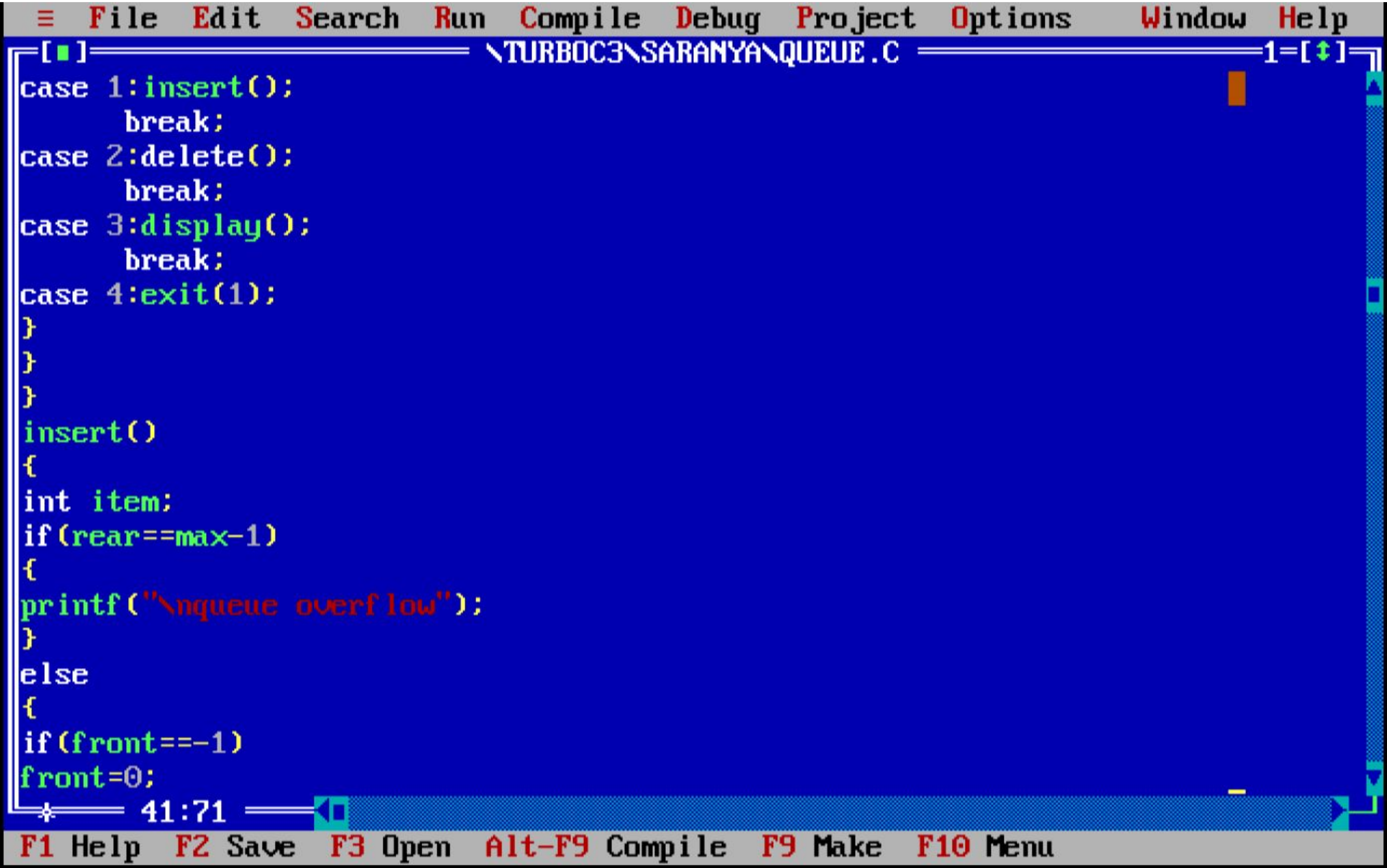


≡ File Edit Search Run Compile Debug Project Options Window Help
[■] \TURBOC3\SARANYA\QUEUE.C 1=[↕]
1:71
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

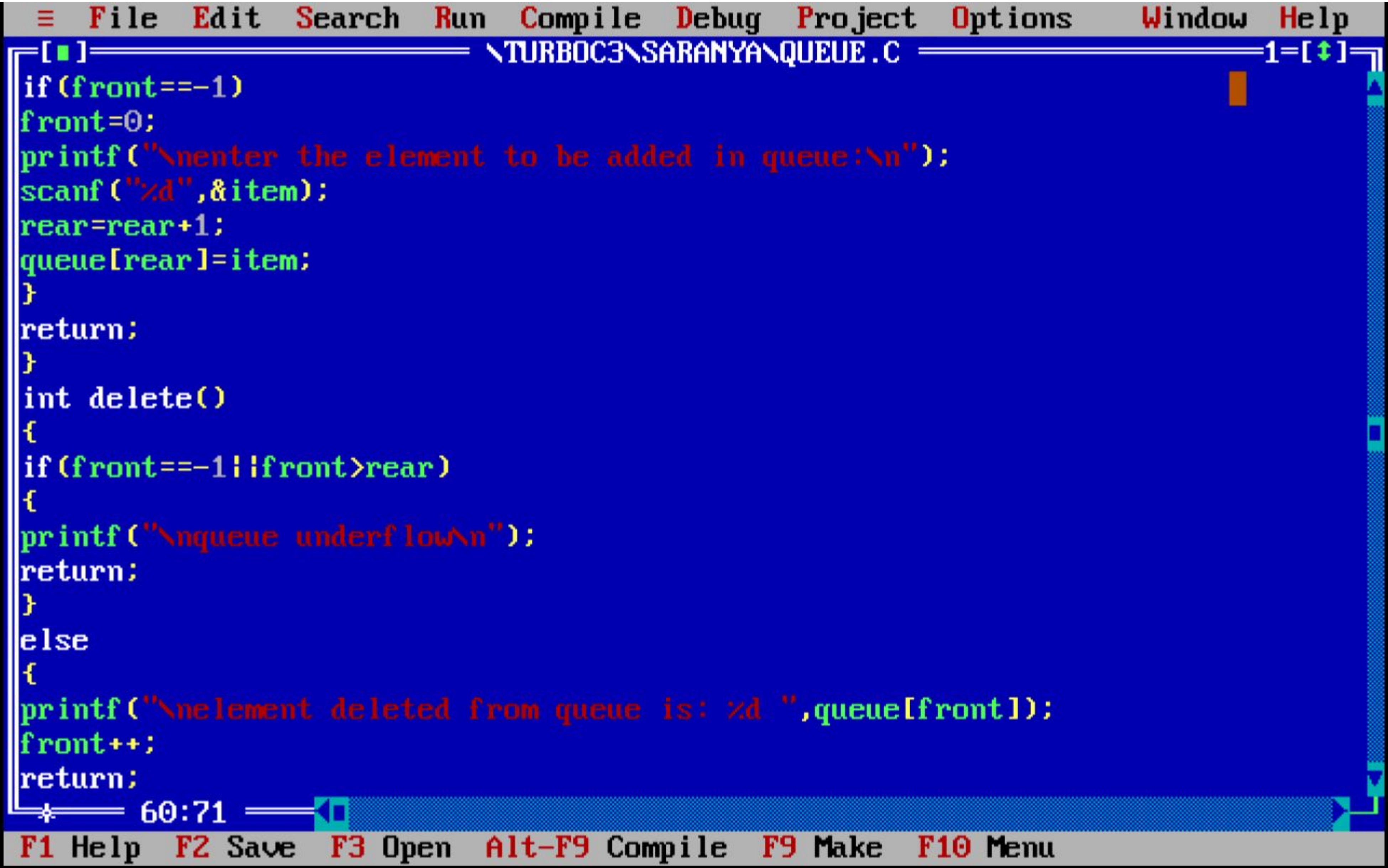
```
#include<stdio.h>
#define max 5
int queue[max],front=-1,rear=-1;
int main()
{
int choice;
clrscr();
while(1)
{
printf("\nqueue operations\n");
printf("\nmain menu:");
printf("\n----------:");
printf("\n1. insert");
printf("\n2. delete");
printf("\n3. display");
printf("\n4. exit");
printf("\n enter your choice:");
scanf("%d",&choice);
switch(choice)
{
case 1:insert();
```



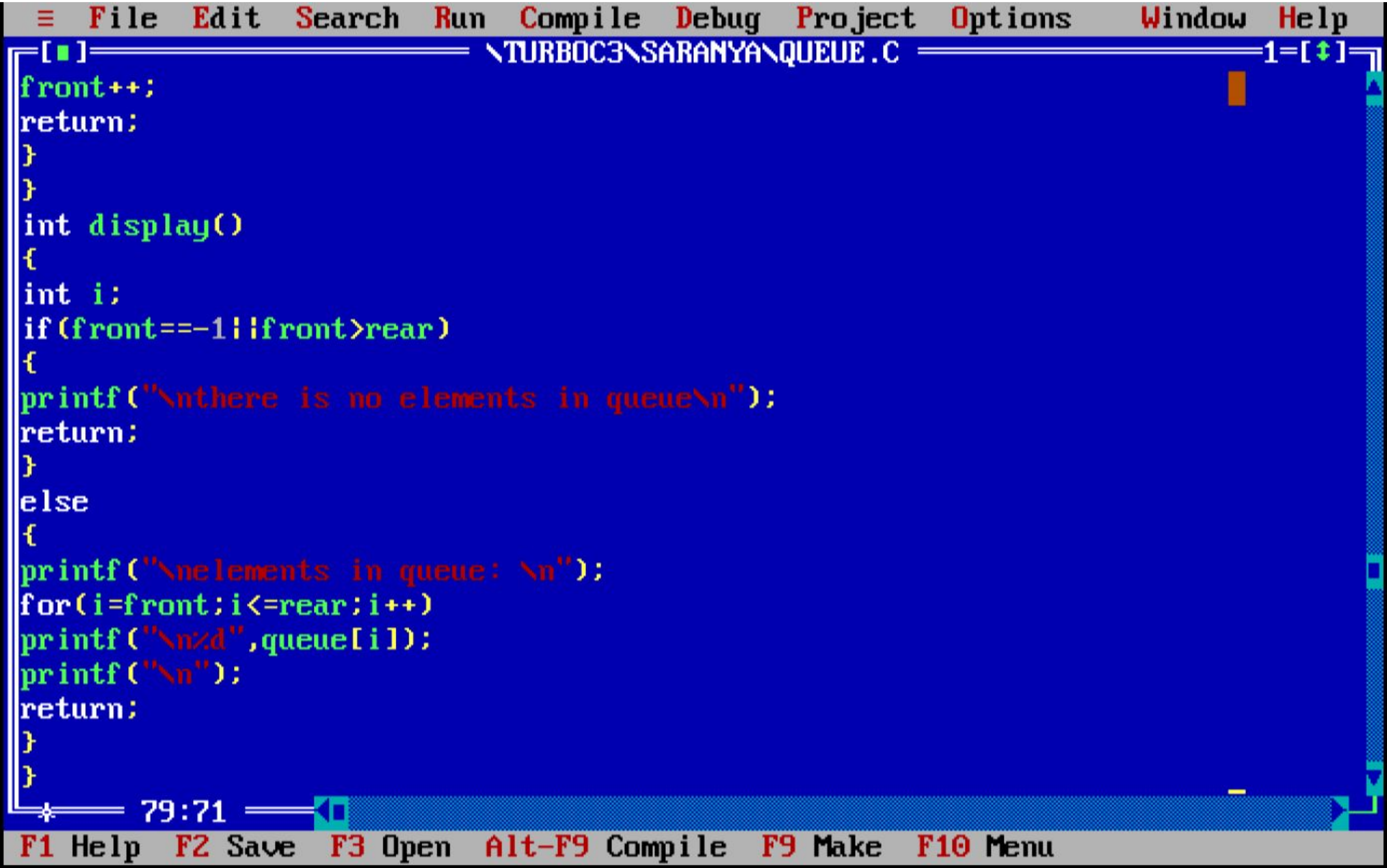
≡ File Edit Search Run Compile Debug Project Options Window Help
[■] \TURBOC3\SARANYA\QUEUE.C 1=[↕]

```
case 1:insert();
       break;
case 2:delete();
       break;
case 3:display();
       break;
case 4:exit(1);
}
}
}
insert()
{
int item;
if(rear==max-1)
{
printf("\nqueue overflow");
}
else
{
if(front==-1)
front=0;
```

41:71
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

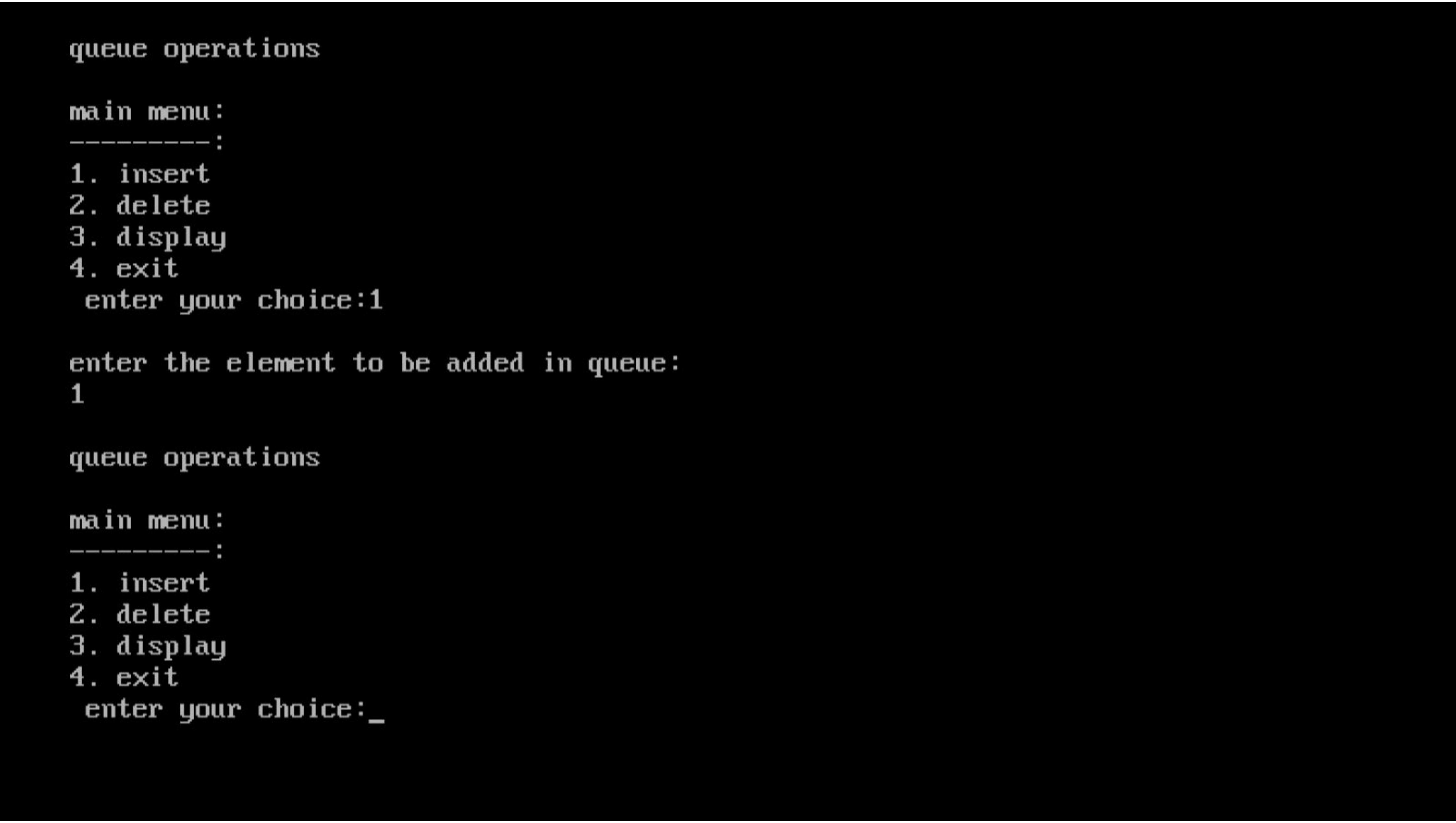

≡ File Edit Search Run Compile Debug Project Options Window Help
[■] \TURBOC3\SARANYA\QUEUE.C 1=[↕]
60:71
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

```
if(front==-1)
front=0;
printf("\nenter the element to be added in queue:\n");
scanf("%d",&item);
rear=rear+1;
queue[rear]=item;
}
return;
}
int delete()
{
if(front==-1||front>rear)
{
printf("\nqueue underflow\n");
return;
}
else
{
printf("\nelement deleted from queue is: %d ",queue[front]);
front++;
return;
```

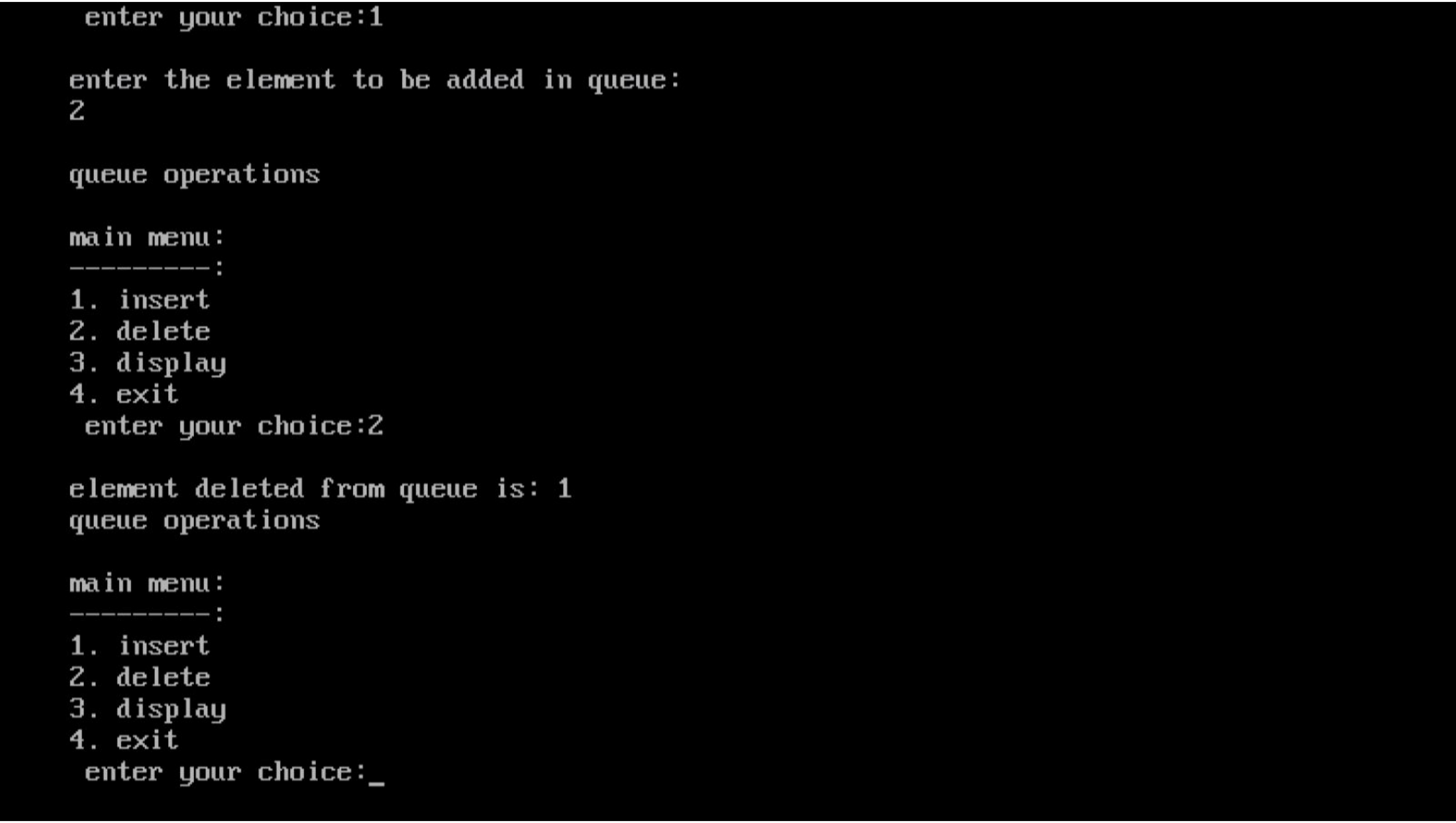


≡ File Edit Search Run Compile Debug Project Options Window Help
\TURBOC3\SARANYA\QUEUE.C
1
79:71
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

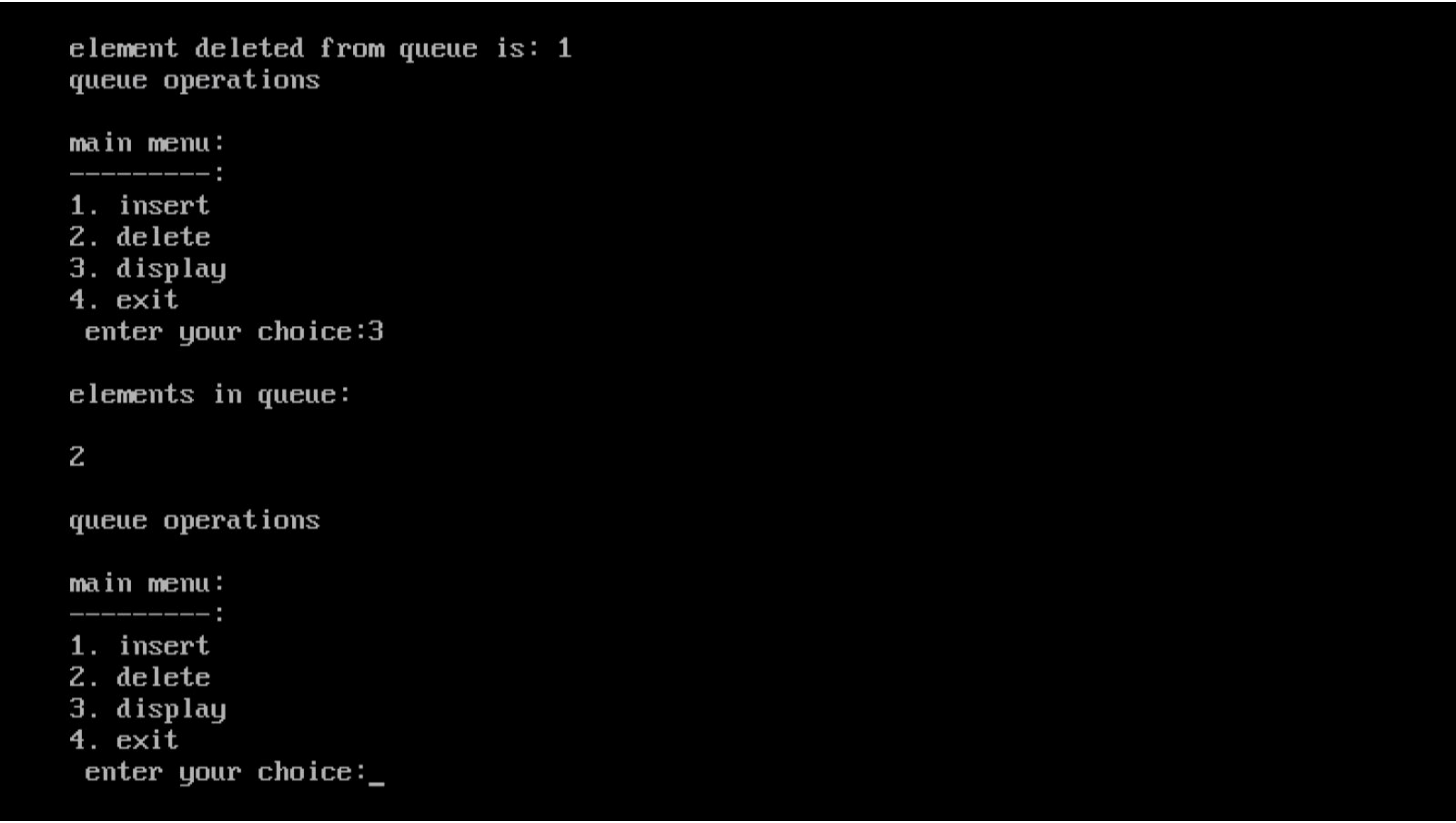
```
front++;
return;
}
}
int display()
{
int i;
if(front==-1||front>rear)
{
printf("\nthere is no elements in queue\n");
return;
}
else
{
printf("\nelements in queue: \n");
for(i=front;i<=rear;i++)
printf("\n%d",queue[i]);
printf("\n");
return;
}
}
```



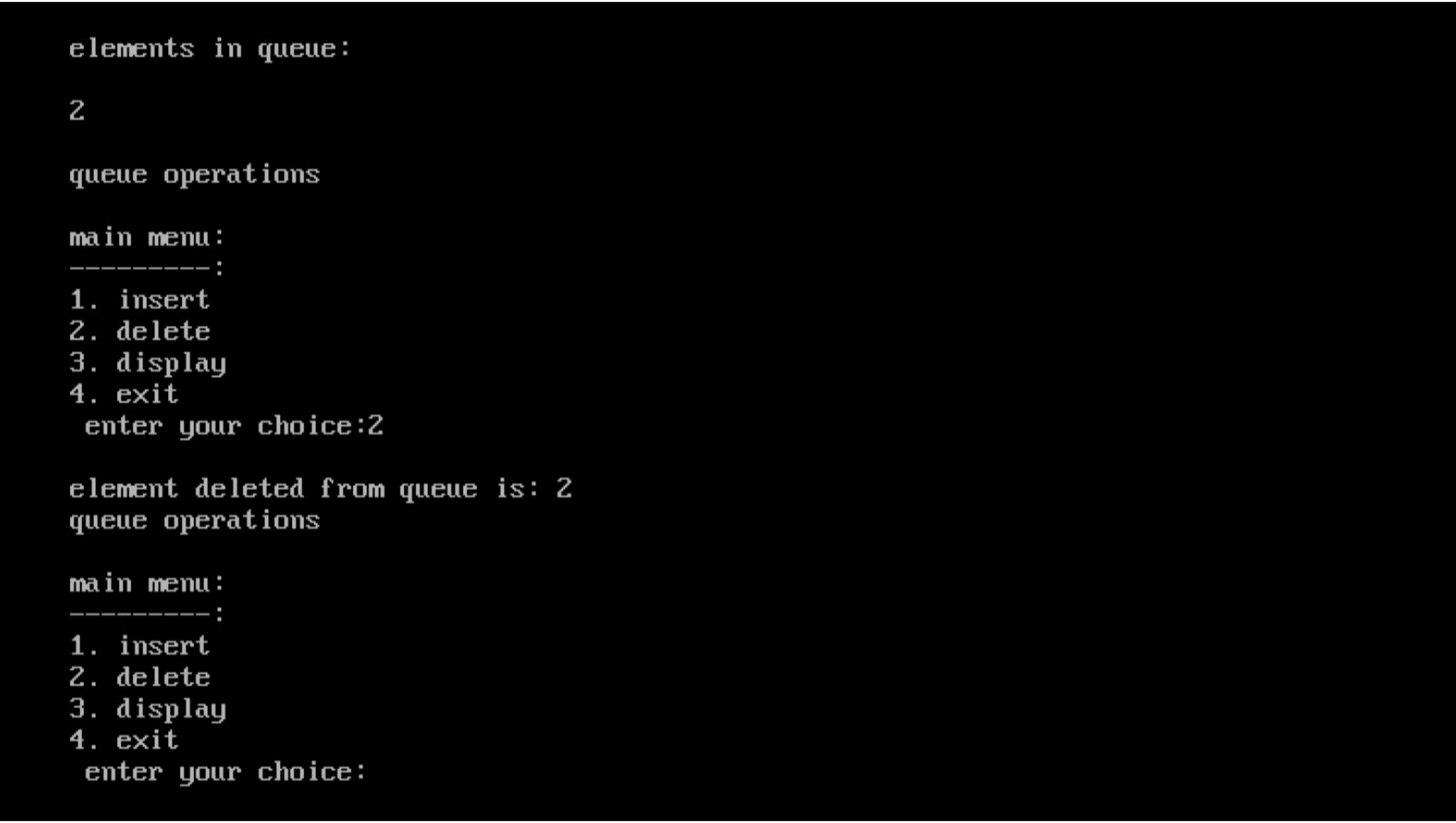
```
queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:1

enter the element to be added in queue:
1

queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:_
```



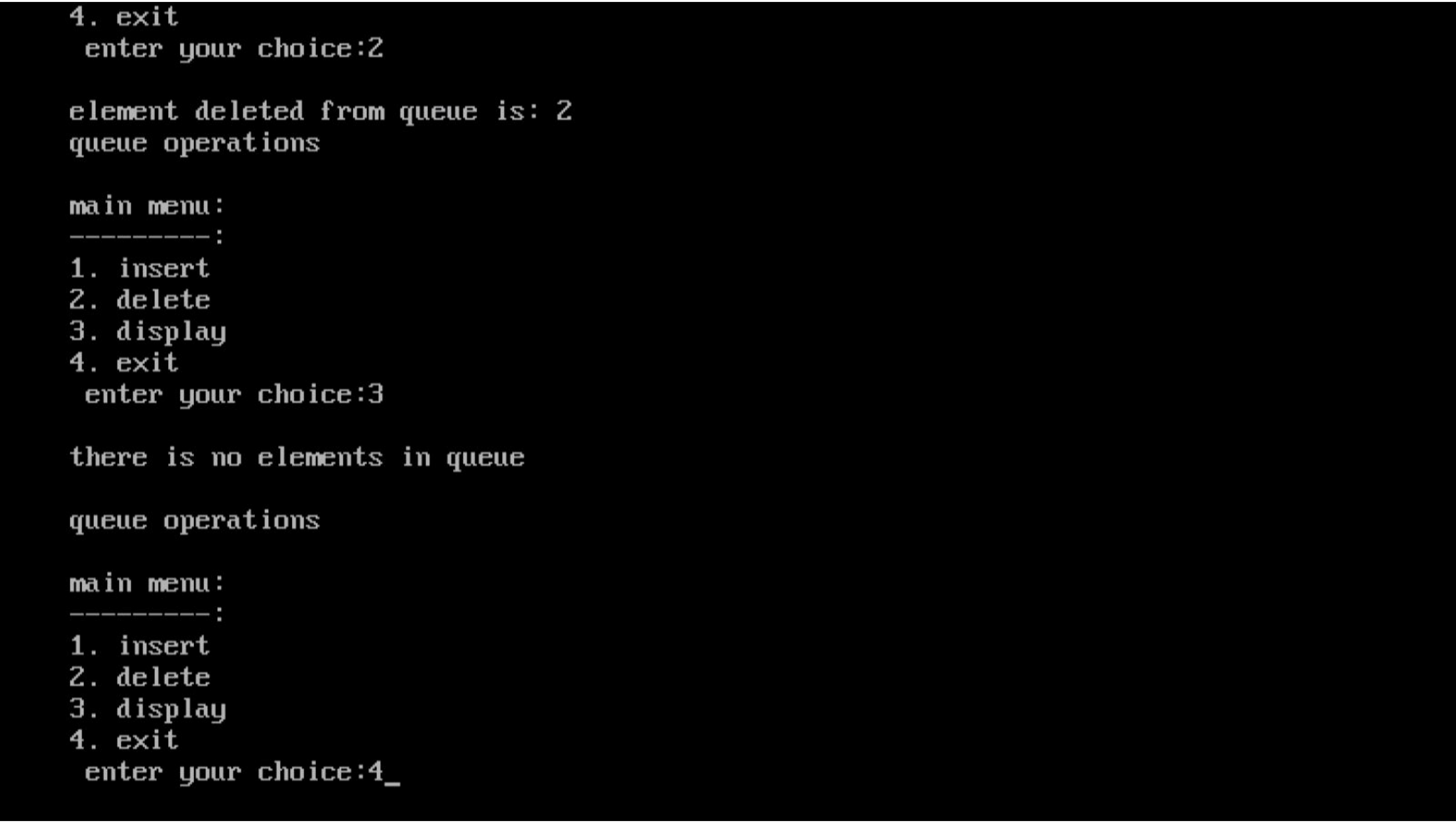
```
 enter your choice:1

enter the element to be added in queue:
2

queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:2

element deleted from queue is: 1
queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:_
```



```
element deleted from queue is: 1
queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:3

elements in queue:

2

queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:_
```



```
elements in queue:

2

queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:2

element deleted from queue is: 2
queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:
```



```
4. exit
 enter your choice:2

element deleted from queue is: 2
queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:3

there is no elements in queue

queue operations

main menu:
---------:
1. insert
2. delete
3. display
4. exit
 enter your choice:4_
```

4/1/2020

Program:- 6

Program to implement queue using array.

Program

```
#include<stdio.h>
#define max 5
int queue[max], front=-1, rear=-1;
int main()
{
 int choice
clrscr();
while(1)
{
 printf("\nqueue operations\n");
 printf("\n main menu:");
 printf("\n- - --:");
printf("\n1. insert");
printf("\n2. Delete");
printf("\n3. display");
printf("\n4. Exit");
printf("\nenter your choice:");
 scanf("%d", &choice);
 switch(choice)
  {
    case1: insert();
            break;
    case2: delete();
            break;
   case 3: display();
            break;
  case 4: exit(0);
  }
 }
}
 insert()
 { int item;
 if(rear == max-1)
  {
   printf("\nqueue overflow");
  }
 else
{
 if(front==-1)
 front=0;
 printf("\nenter the element to be added in queue:\n");
 scanf("%d", &item);
```

```
  rear = rear +1;
  queue[rear] = item;
}
 return;
 }
 int delete()
  {
   if(front == -1 || front > rear)
    {
     printf("\nqueue underflow\n");
    return
    }
    else
    {
   printf("\nelement deleted from queue is: %d",
                                        queue[front]);
   front++;
   return;
   }
  }
  int display()
   {
      int i;
     if(front == -1 || front > rear)
```

```
{
  printf("\n there is no elements in queue \n");
 return;
}
else
{
printf("\nelements in queue: \n");
for(i=front; i<=rear; i++)
printf("\n %d", queue[i]);
printf("\n");
return;
 }
}
```

output

```
queue operations

Main menu:
- - - - -

1. insert
2. delete
3. display
4. exit

enter your choice: 2

~~element deleted from queue is 1~~
enter the element to be added in queue!
queue operations

main menu
- - - -

1. insert
2. delete
3. display
4. exit
enter your choice : 1
~~element~~ enter the element to be added in queue!
2
queue operation
main menu
- - - -
```

```
1. Insert
2. delete
3. display
4. exit
enter your choice: 2
element deleted from queue is:1
queue operations
main menu
- - - -
1. Insert
2. delete
3. display
4. exit
enter your choice: 3
elements in queue:
2
queue operations
main menu
- - - -
1. Insert
2. delete
3. display
4. exit
enter your choice: 2
```

```
element deleted from queue is: 2
queue operations
main menu
- - - -
1. insert
2. delete
3. display
4. exit
enter your choice : 3
there is no element in queue
queue operation
main menu:
- - - -
1. insert
2. delete
3. display
4. exit
enter your choice: 4
```