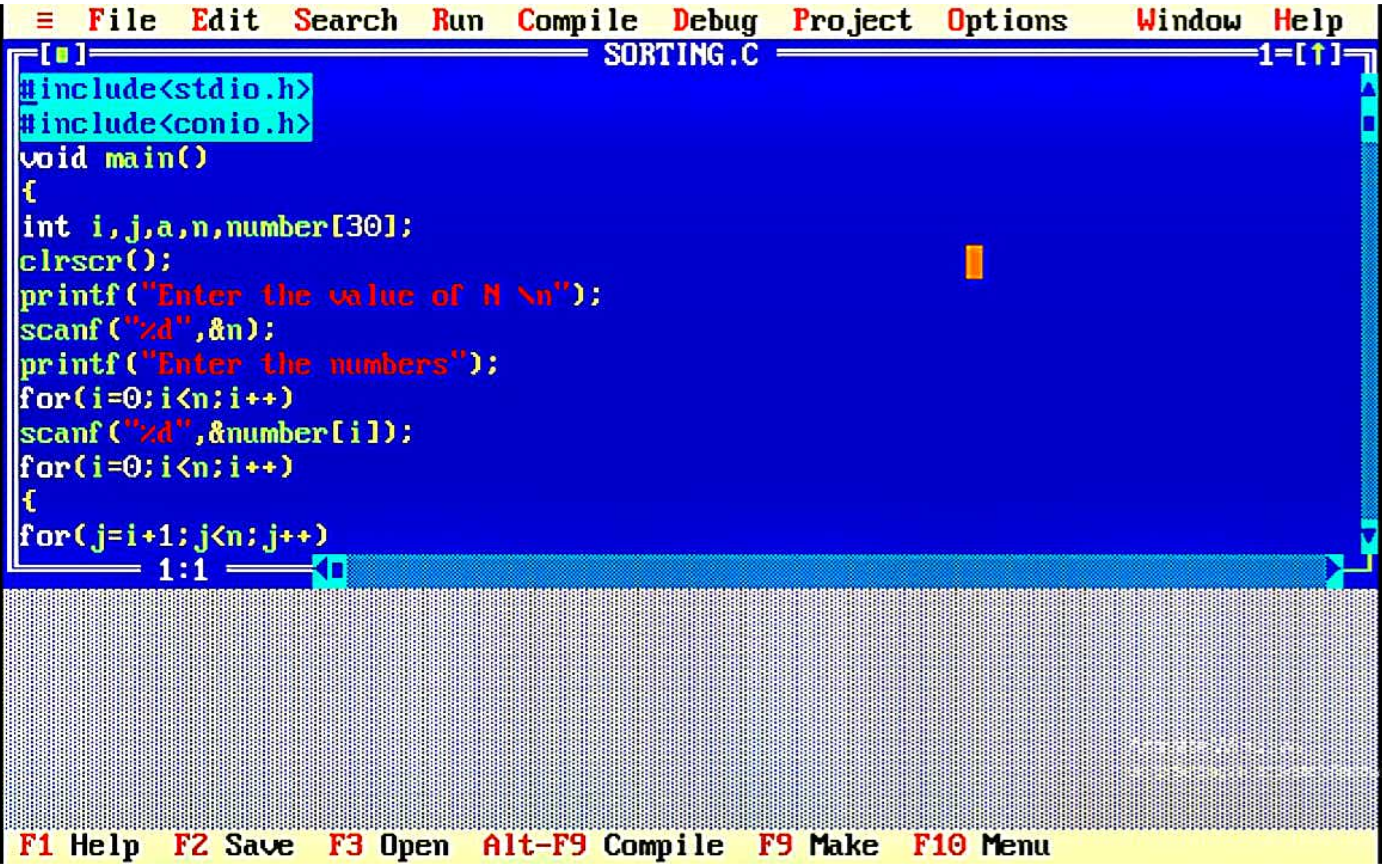
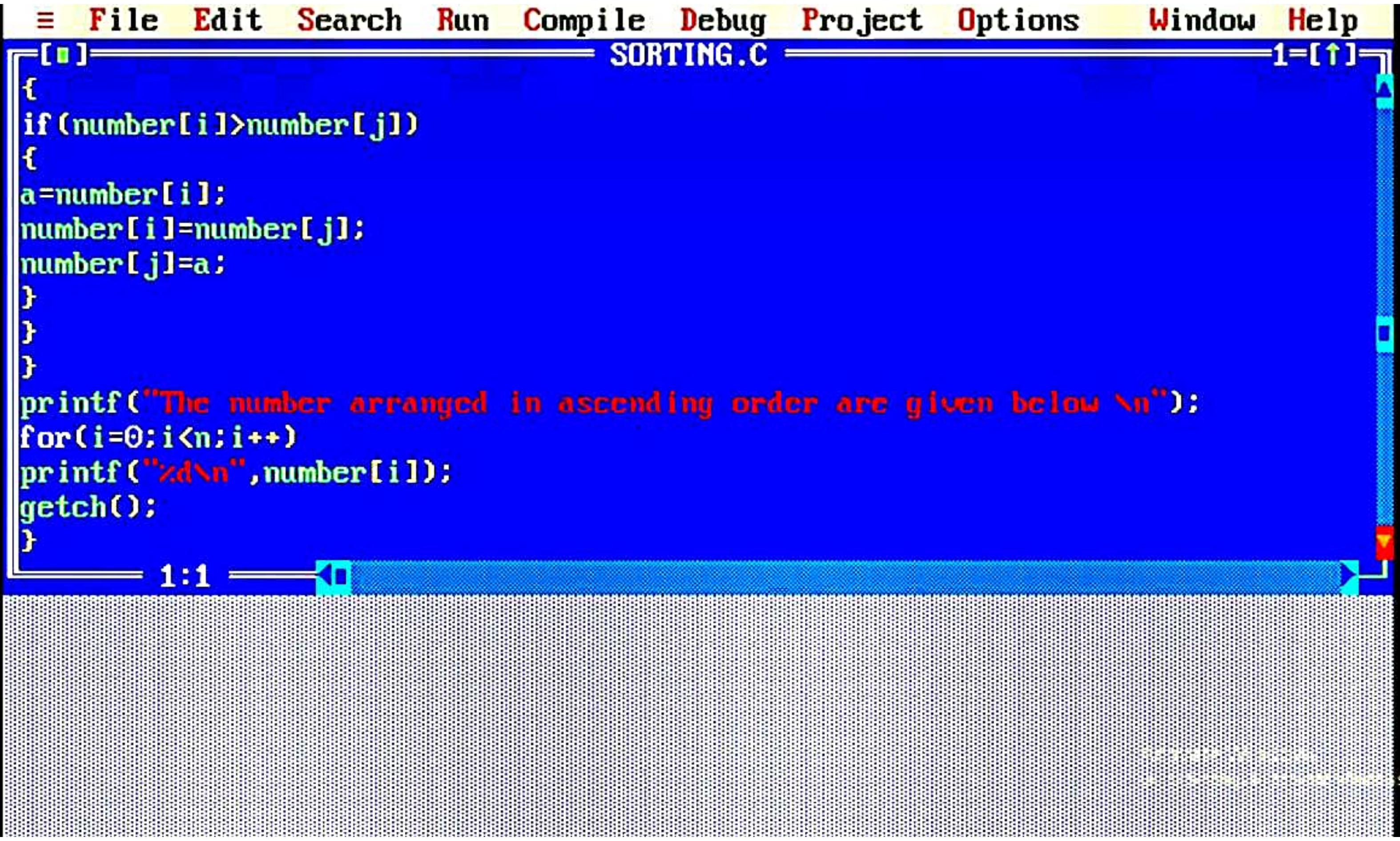
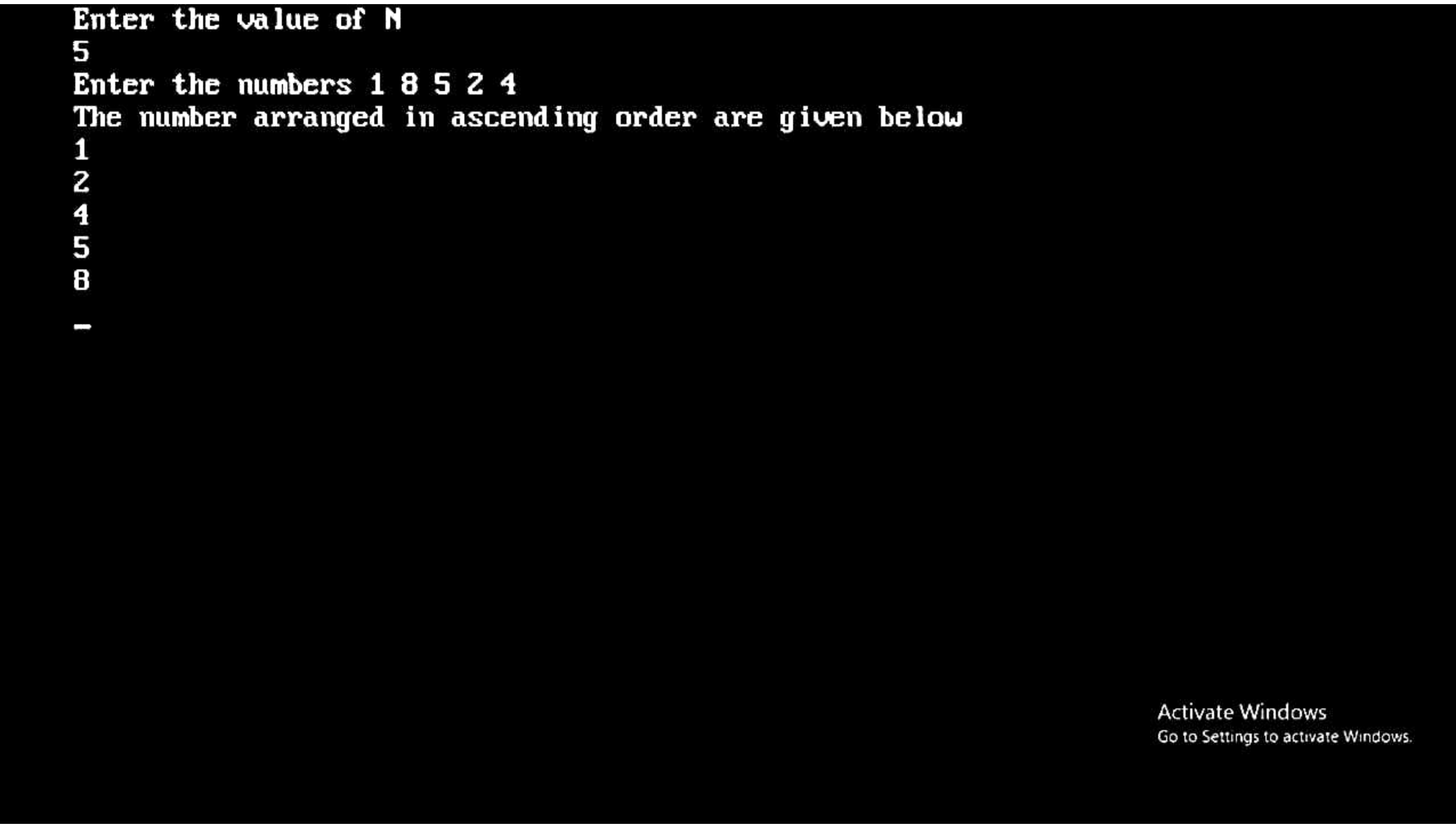

≡ File Edit Search Run Compile Debug Project Options Window Help
[■]
SORTING.C
1=[↑]
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

```c
#include<stdio.h>
#include<conio.h>
void main()
{
int i,j,a,n,number[30];
clrscr();
printf("Enter the value of N \n");
scanf("%d",&n);
printf("Enter the numbers");
for(i=0;i<n;i++)
scanf("%d",&number[i]);
for(i=0;i<n;i++)
{
for(j=i+1;j<n;j++)
```

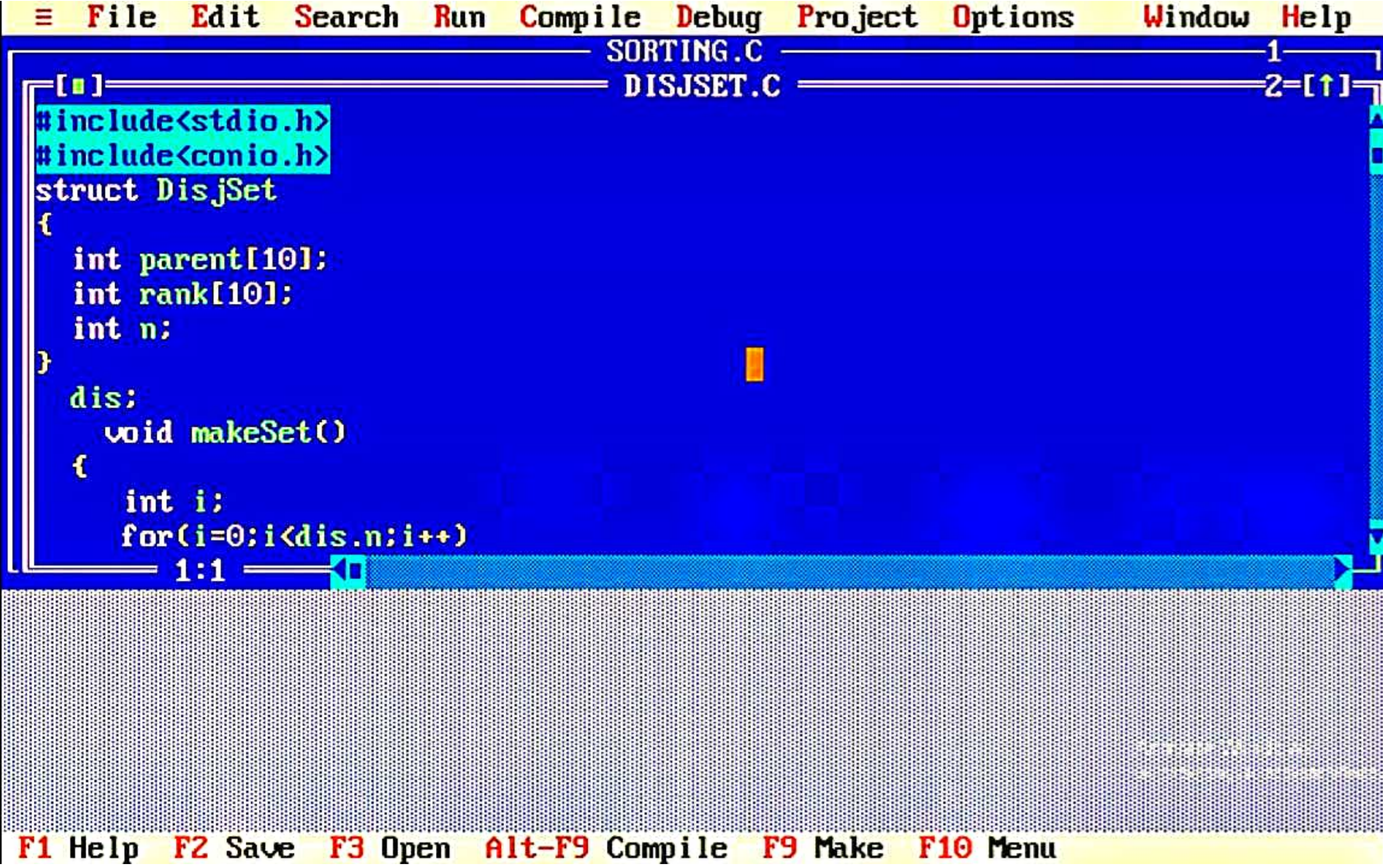


≡ File Edit Search Run Compile Debug Project Options Window Help
[■]
SORTING.C
1=[↑]
1:1

```
{
if(number[i]>number[j])
{
a=number[i];
number[i]=number[j];
number[j]=a;
}
}
}
printf("The number arranged in ascending order are given below \n");
for(i=0;i<n;i++)
printf("%d\n",number[i]);
getch();
}
```

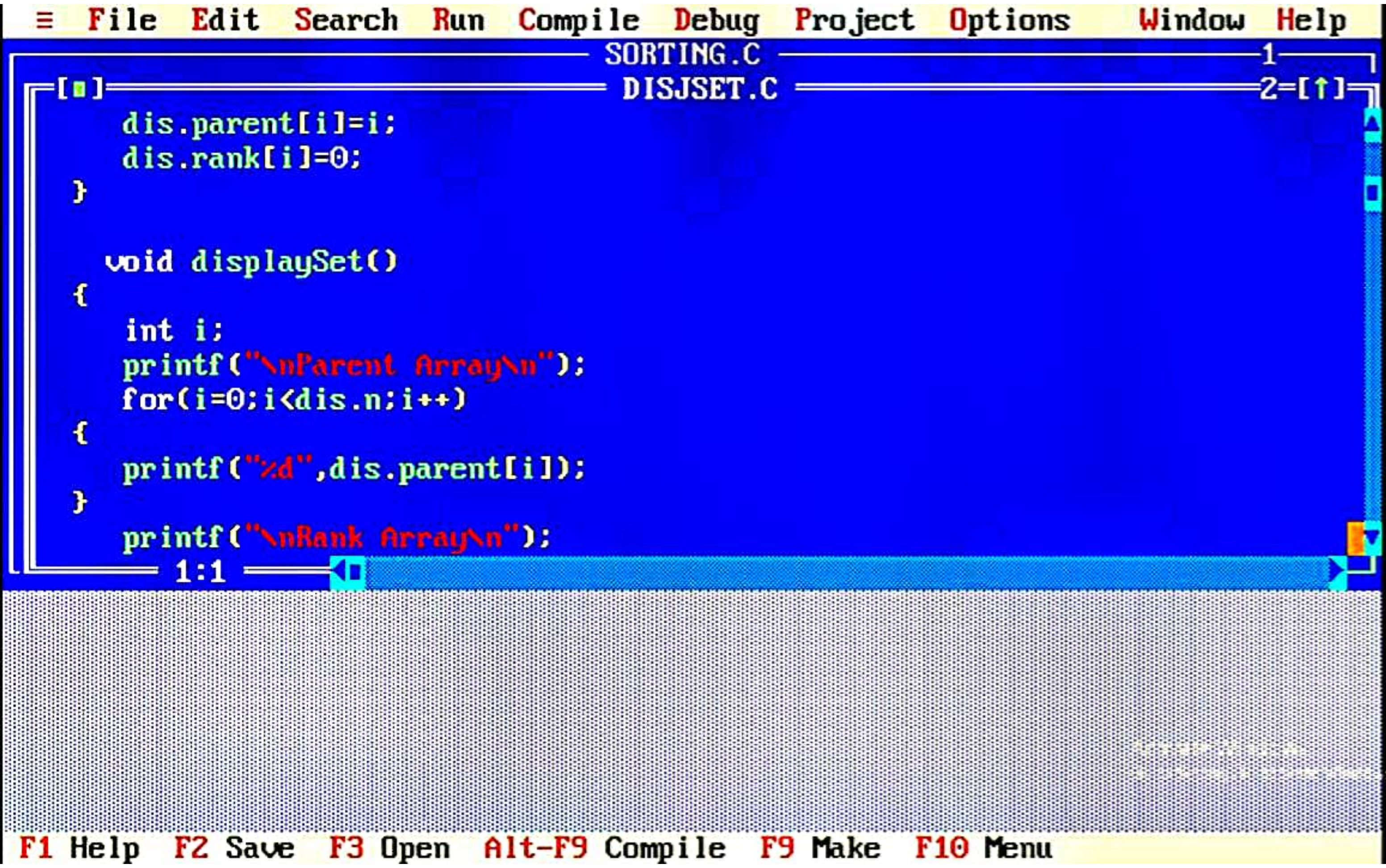


```
Enter the value of N
5
Enter the numbers 1 8 5 2 4
The number arranged in ascending order are given below
1
2
4
5
8

_
```

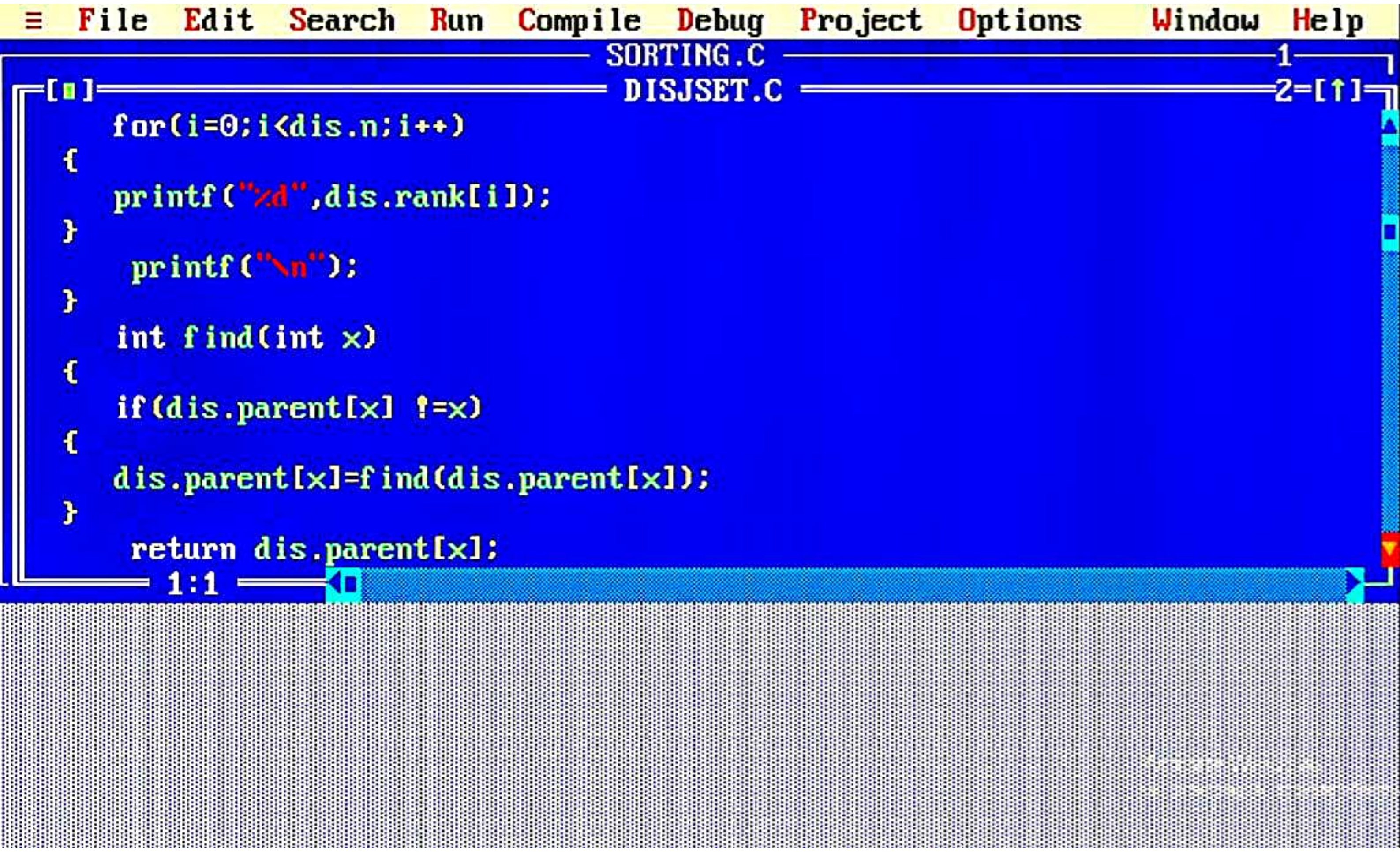


≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2
[↑]
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

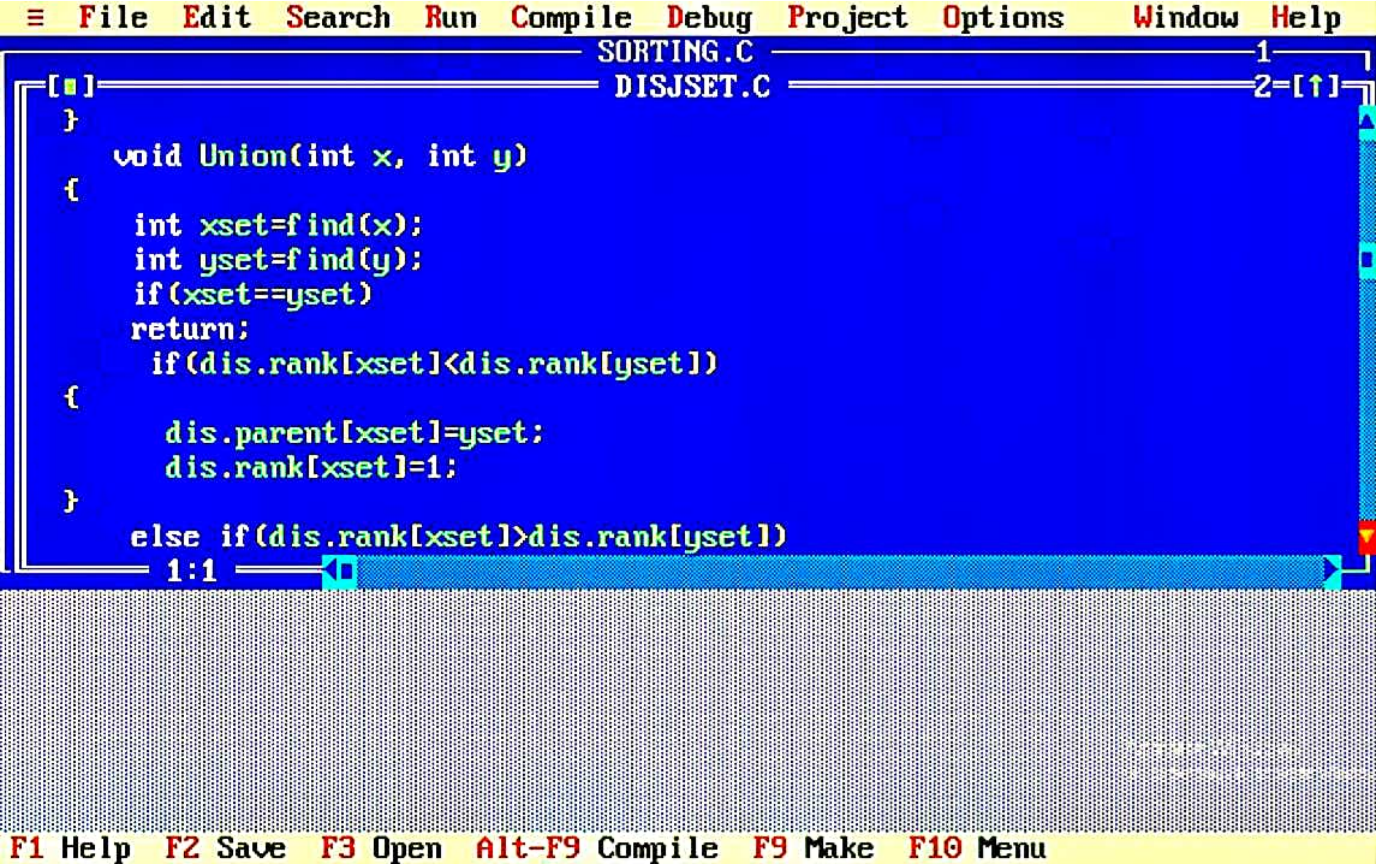
```
#include<stdio.h>
#include<conio.h>
struct DisjSet
{
  int parent[10];
  int rank[10];
  int n;
}
  dis;
    void makeSet()
  {
     int i;
     for(i=0;i<dis.n;i++)
```


≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2=[↑]
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

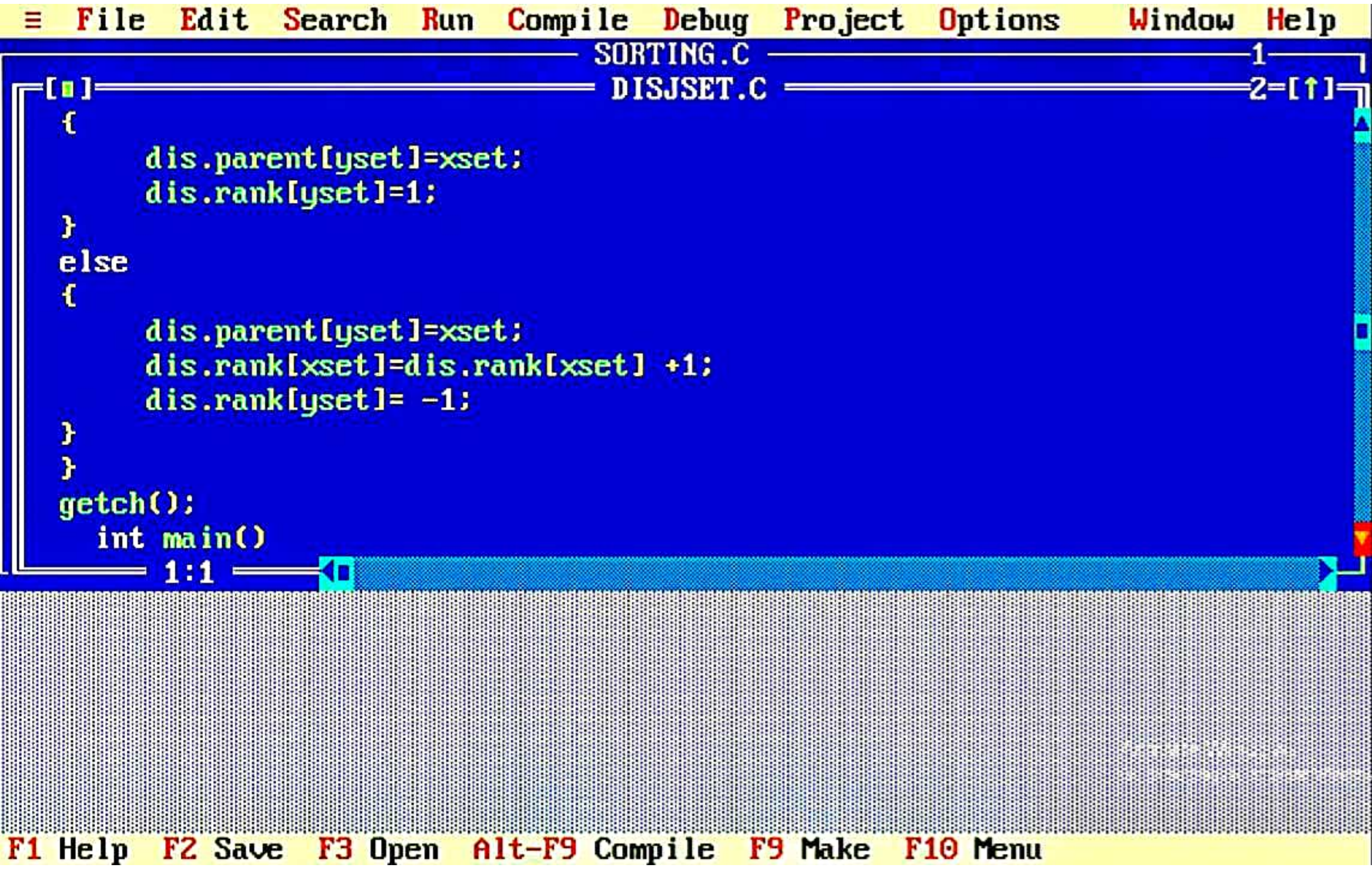
```
    dis.parent[i]=i;
    dis.rank[i]=0;
 }

   void displaySet()
 {
    int i;
    printf("\nParent Array\n");
    for(i=0;i<dis.n;i++)
 {
    printf("%d",dis.parent[i]);
 }
    printf("\nRank Array\n");
```


≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2=[↑]
1:1

```
    for(i=0;i<dis.n;i++)
  {
    printf("%d",dis.rank[i]);
  }
     printf("\n");
  }
    int find(int x)
  {
    if(dis.parent[x] !=x)
  {
    dis.parent[x]=find(dis.parent[x]);
  }
     return dis.parent[x];
```

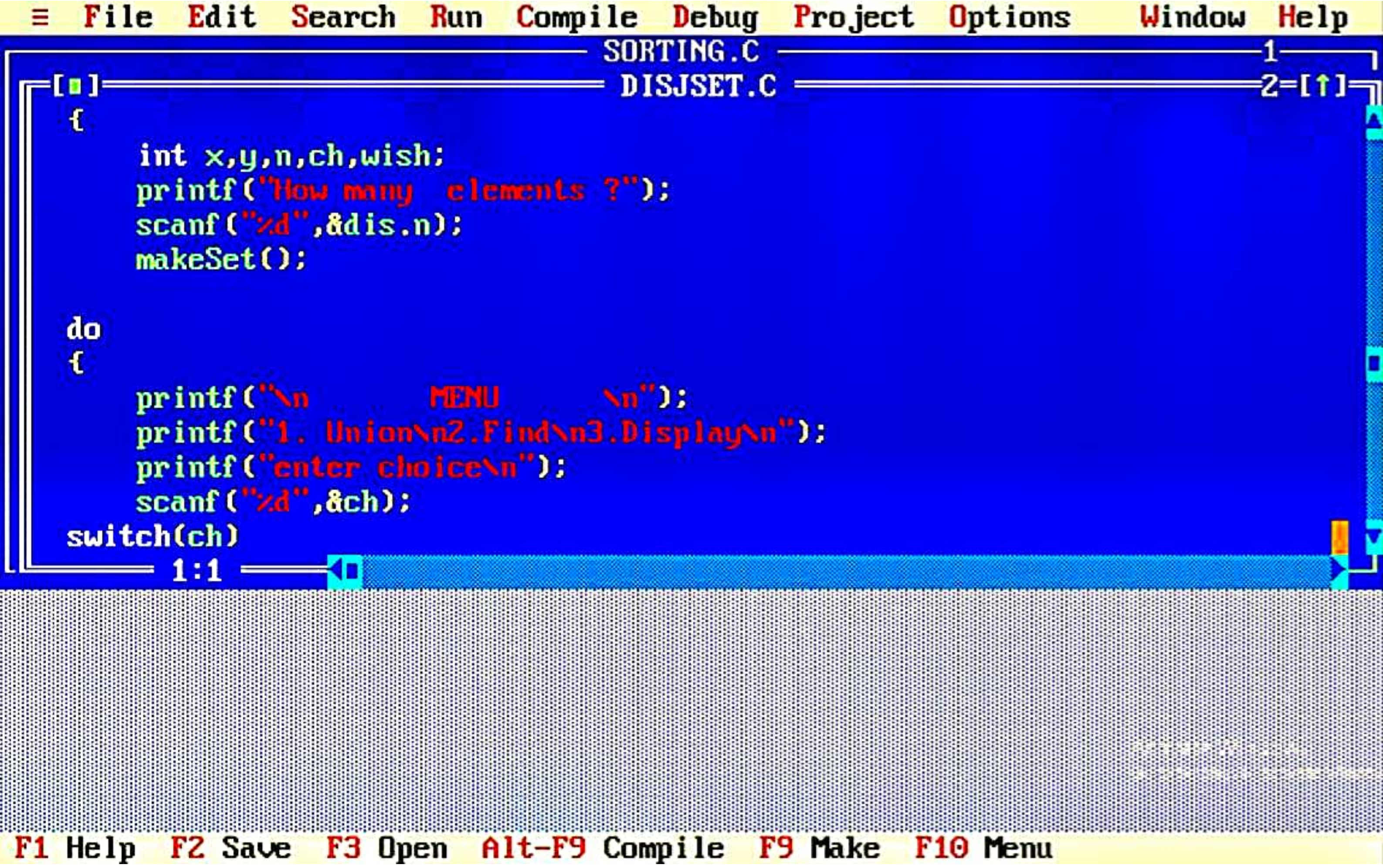

≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2 [↑]
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

```
}
   void Union(int x, int y)
{
    int xset=find(x);
    int yset=find(y);
    if(xset==yset)
    return;
     if(dis.rank[xset]<dis.rank[yset])
{
      dis.parent[xset]=yset;
      dis.rank[xset]=1;
}
    else if(dis.rank[xset]>dis.rank[yset])
```



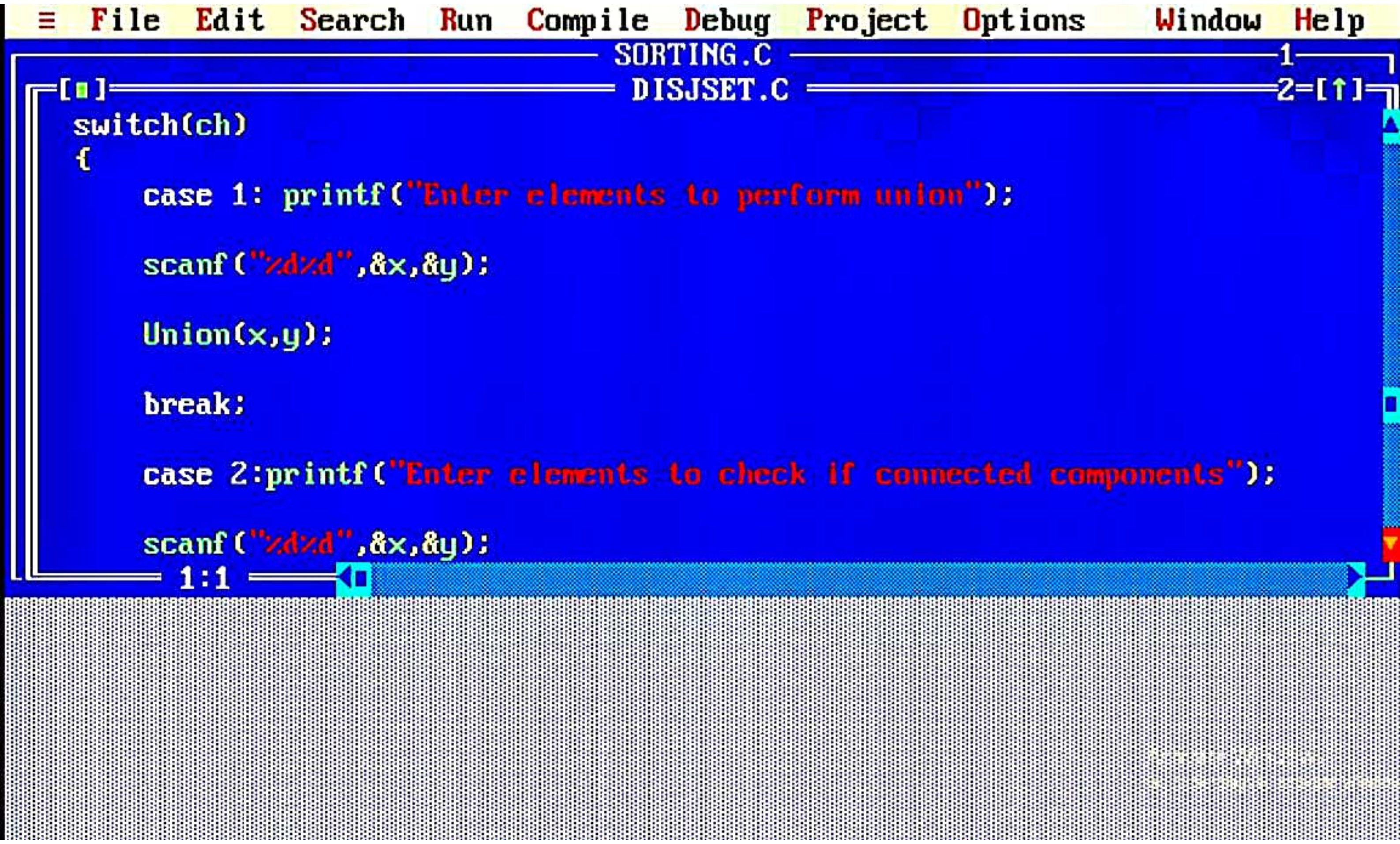
≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2
[↑]
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

```
 {
      dis.parent[yset]=xset;
      dis.rank[yset]=1;
 }
 else
 {
      dis.parent[yset]=xset;
      dis.rank[xset]=dis.rank[xset] +1;
      dis.rank[yset]= -1;
 }
 }
 getch();
   int main()
```

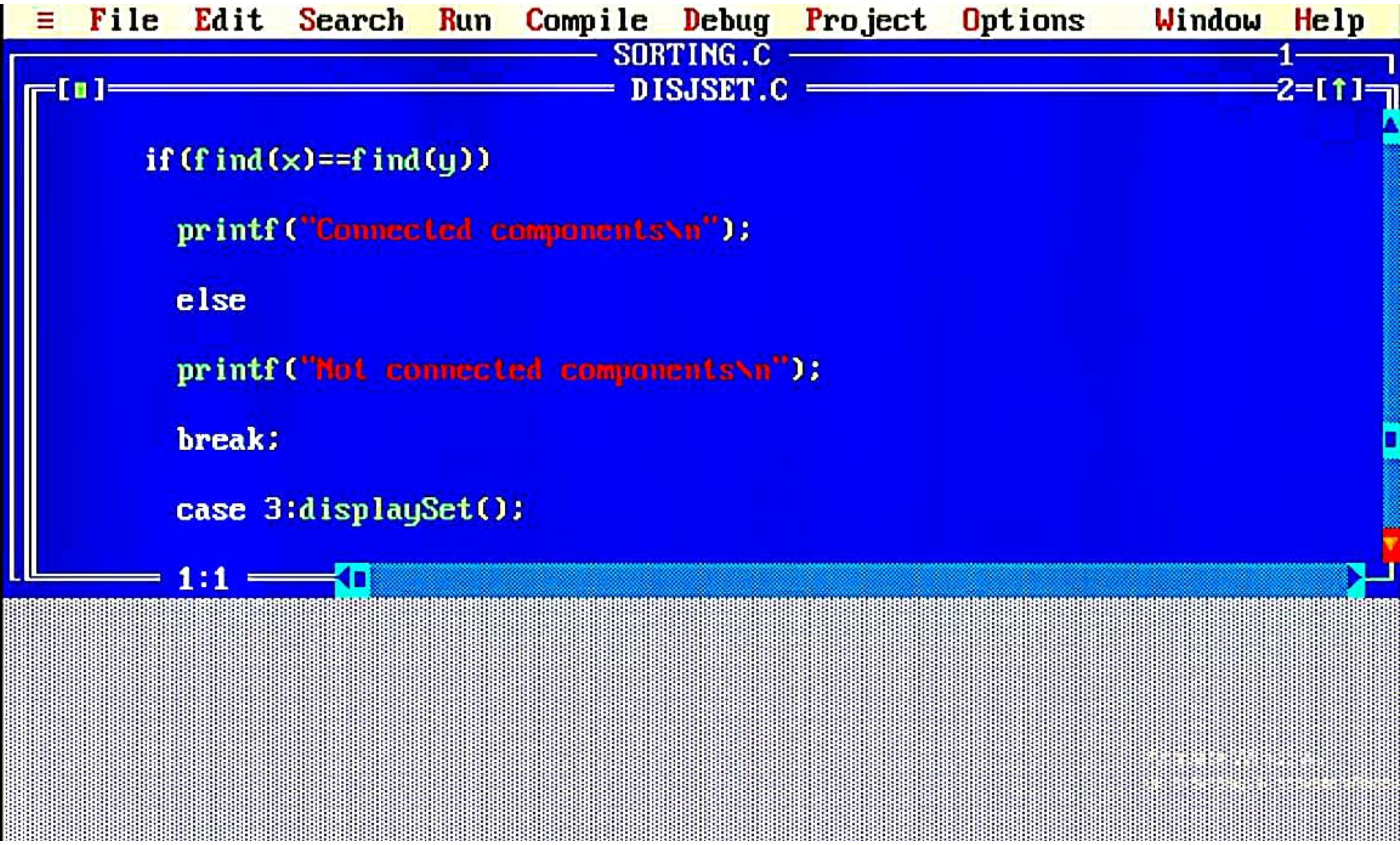


≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2=[↑]
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

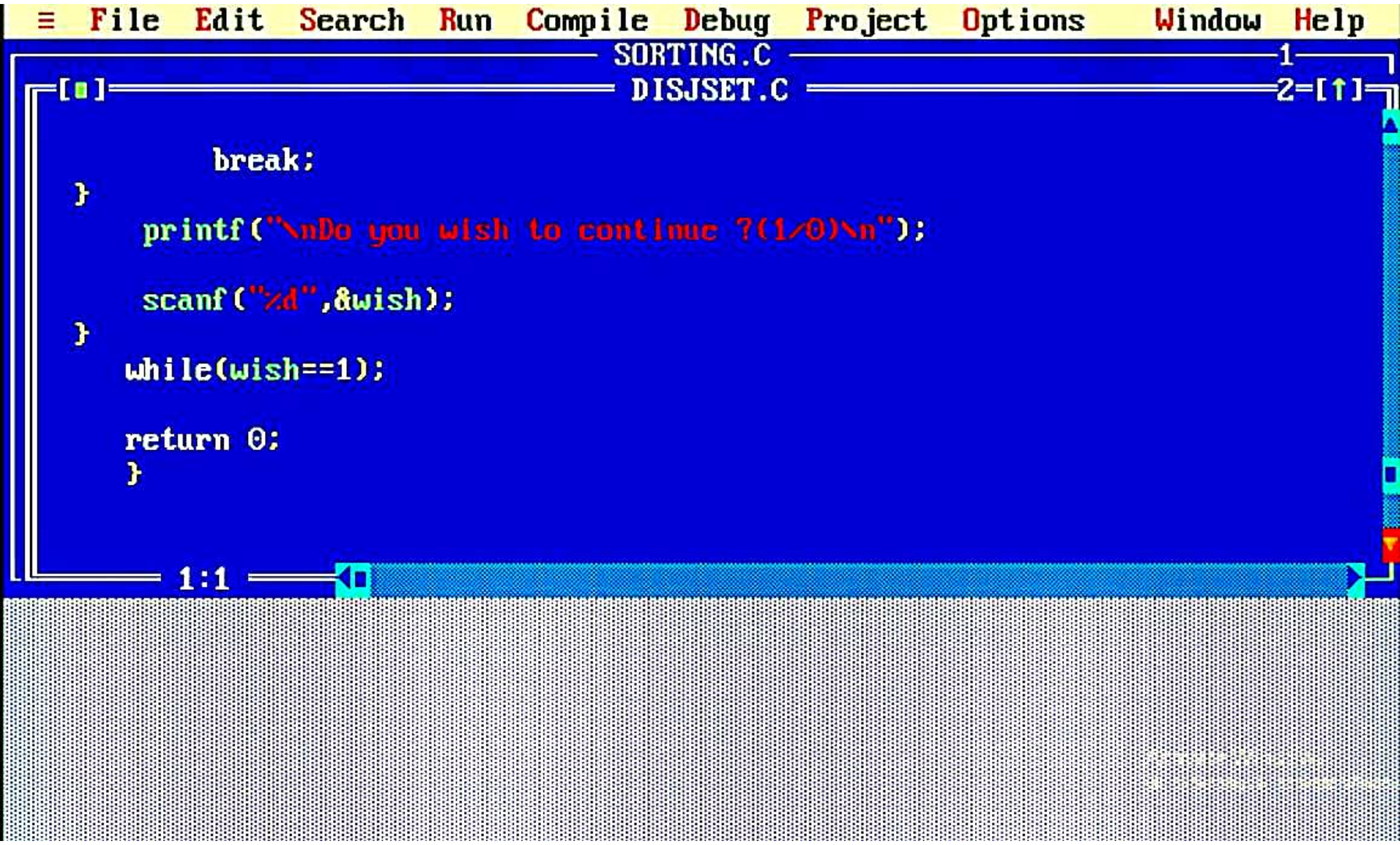
```
{
    int x,y,n,ch,wish;
    printf("How many  elements ?");
    scanf("%d",&dis.n);
    makeSet();


do
{
    printf("\n       MENU       \n");
    printf("1. Union\n2.Find\n3.Display\n");
    printf("enter choice\n");
    scanf("%d",&ch);
switch(ch)
```


≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2=[↑]
1:1

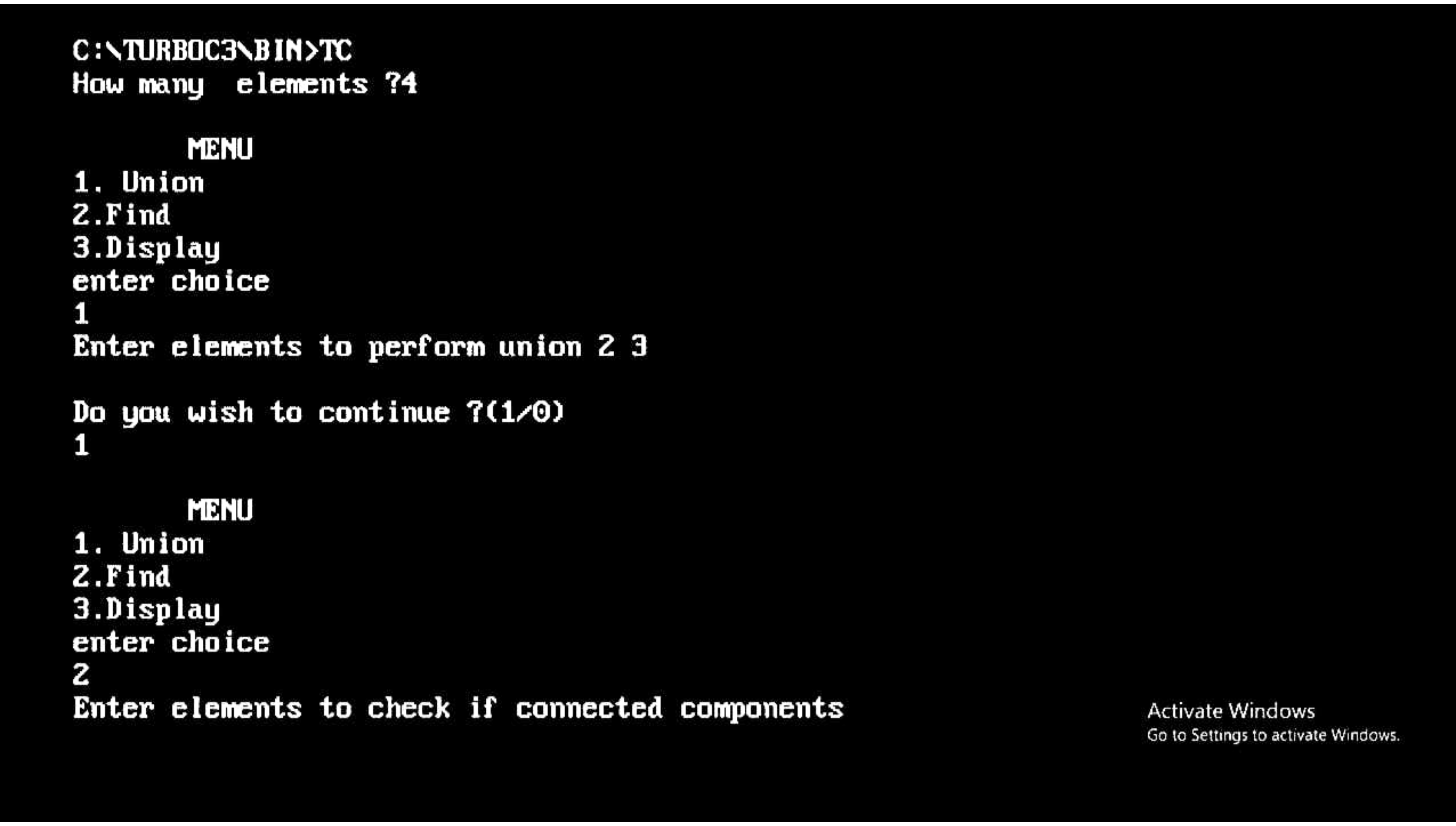
```
switch(ch)
{
    case 1: printf("Enter elements to perform union");

    scanf("%d%d",&x,&y);

    Union(x,y);

    break;

    case 2:printf("Enter elements to check if connected components");

    scanf("%d%d",&x,&y);
```


≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[▪]
DISJSET.C
2=[↑]
1:1

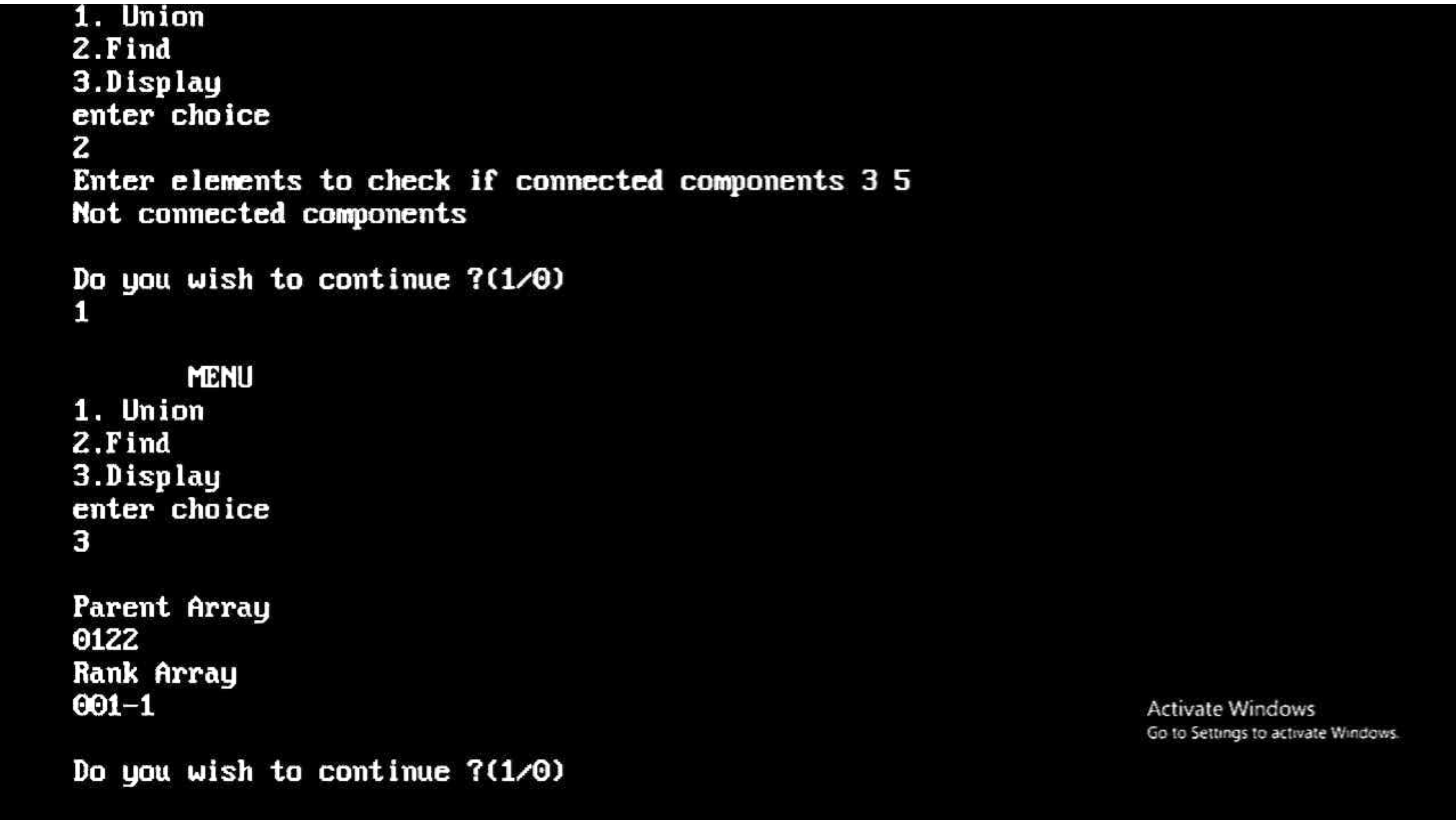
```
    if(find(x)==find(y))
      printf("Connected components\n");
      else
      printf("Not connected components\n");
      break;
      case 3:displaySet();
```


≡ File Edit Search Run Compile Debug Project Options Window Help
SORTING.C
1
[■]
DISJSET.C
2=[↑]
1:1

```
        break;
 }
     printf("\nDo you wish to continue ?(1/0)\n");

     scanf("%d",&wish);
 }
    while(wish==1);

    return 0;
    }
```



```
C:\TURBOC3\BIN>TC
How many  elements ?4

       MENU
1. Union
2.Find
3.Display
enter choice
1
Enter elements to perform union 2 3

Do you wish to continue ?(1/0)
1

       MENU
1. Union
2.Find
3.Display
enter choice
2
Enter elements to check if connected components
```



```
1. Union
2.Find
3.Display
enter choice
2
Enter elements to check if connected components 3 5
Not connected components

Do you wish to continue ?(1/0)
1

       MENU
1. Union
2.Find
3.Display
enter choice
3

Parent Array
0122
Rank Array
001-1

Do you wish to continue ?(1/0)
```