VP Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

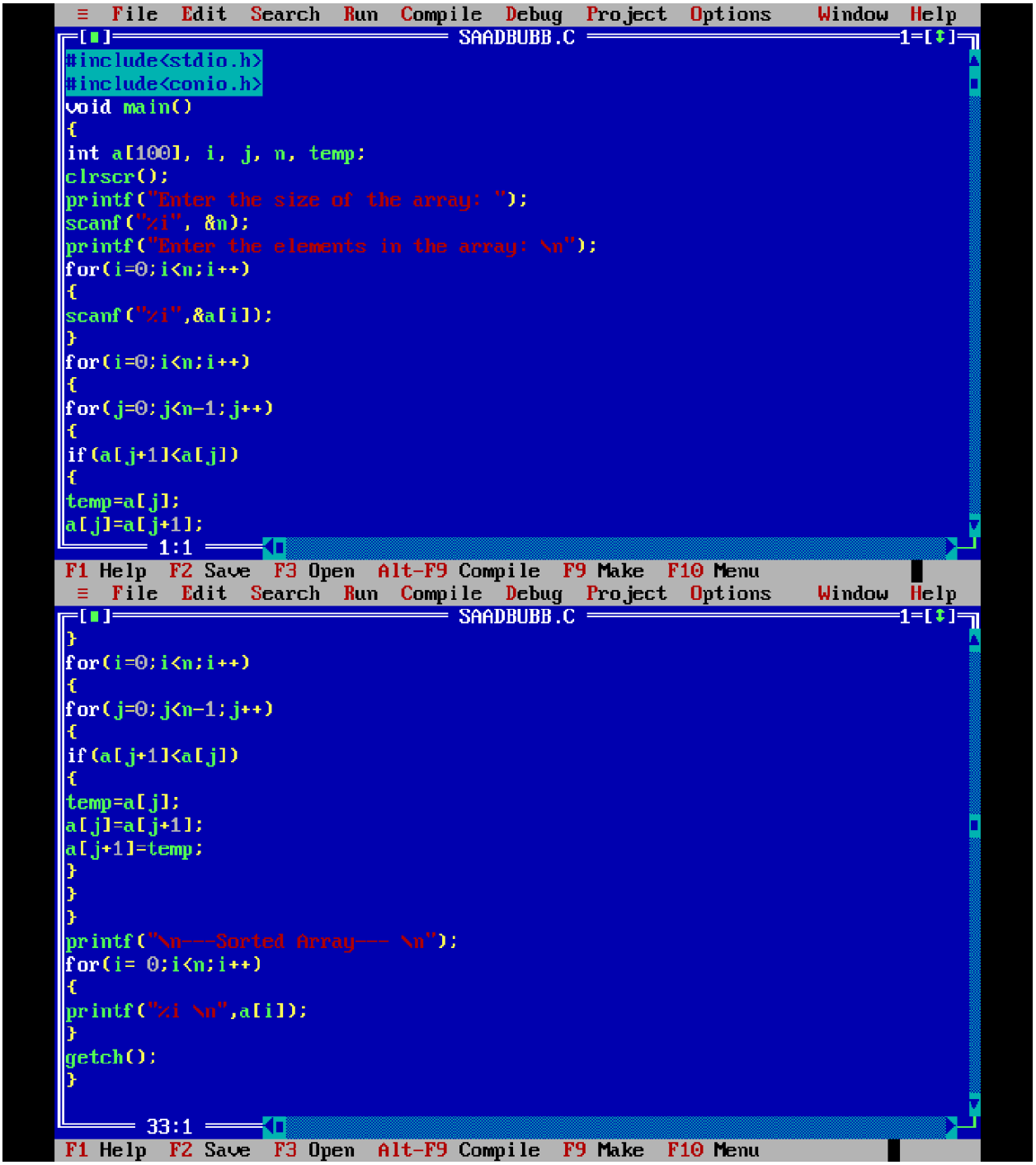
Subject: - DATA STRUCTURE	**Subject Code:** 313301
Semester: - III	**Course:** COMPUTER ENGINEERING
Laboratory No: L003	**Name of Subject Teacher:** Prof. Imraan S.
Name of Student: Saad Sharif Kazi	**Roll Id: -** 24203A0013
Experiment No:	6
Title of Experiment	* Write a 'C' Program to Sort an Array of numbers using Bubble Sort Method.

Aim: Write a 'C' Program to Sort an Array of numbers using Bubble Sort Method.

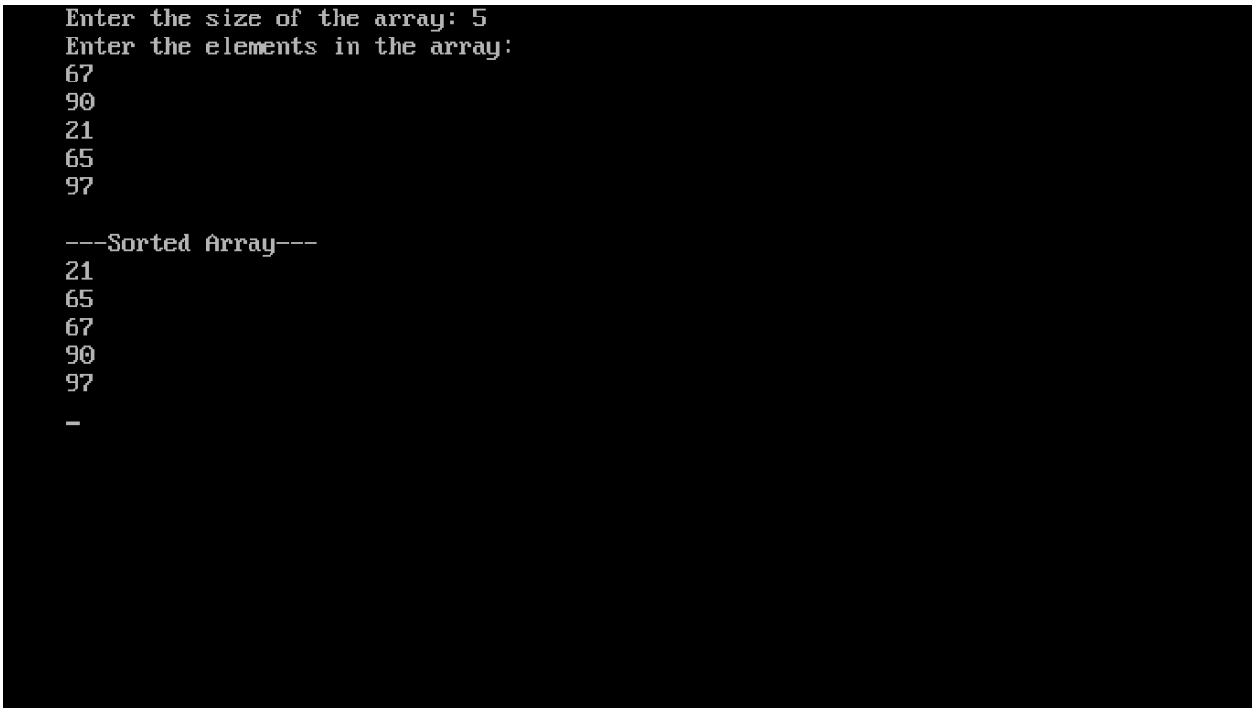
Algorithm:

Step 1: Start
Step 2: Declare an integer array a[100] and variables i, n
Step 3: Clear screen using clrscr()
Step 4: Print "Enter the size of the array"
Step 5: Scan value of n from keyboard
Step 6: Print "Enter the elements in the array"
Step 7: Run a loop from i = 0 to i < n
Step 7.1: Scan each element and store it in a[i]
Step 8: Call the function sort(a, n)
Step 9: Inside the sort() function
Step 9.1: Declare variables i, j, temp
Step 9.2: Run a loop from i = 0 to i < n
Step 9.2.1: Run a nested loop from j = 0 to j < n
Step 9.2.1.1: If a[j+1] < a[j], then
Step 9.2.1.1.1: Swap a[j] and a[j+1] using temp
Step 10: After returning from function, print "Sorted Array"
Step 11: Run a loop from i = 0 to i < n
Step 11.1: Print each element a[i]
Step 12: Stop**Code:**

CODE:


File Edit Search Run Compile Debug Project Options Window Help
SAADBUBB.C
#include<stdio.h>
#include<conio.h>
void main()
{
int a[100], i, j, n, temp;
clrscr();
printf("Enter the size of the array: ");
scanf("%i", &n);
printf("Enter the elements in the array: \n");
for(i=0;i<n;i++)
{
scanf("%i",&a[i]);
}
for(i=0;i<n;i++)
{
for(j=0;j<n-1;j++)
{
if(a[j+1]<a[j])
{
temp=a[j];
a[j]=a[j+1];
1:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu
File Edit Search Run Compile Debug Project Options Window Help
SAADBUBB.C
}
for(i=0;i<n;i++)
{
for(j=0;j<n-1;j++)
{
if(a[j+1]<a[j])
{
temp=a[j];
a[j]=a[j+1];
a[j+1]=temp;
}
}
}
printf("\n---Sorted Array--- \n");
for(i= 0;i<n;i++)
{
printf("%i \n",a[i]);
}
getch();
}
33:1
F1 Help F2 Save F3 Open Alt-F9 Compile F9 Make F10 Menu

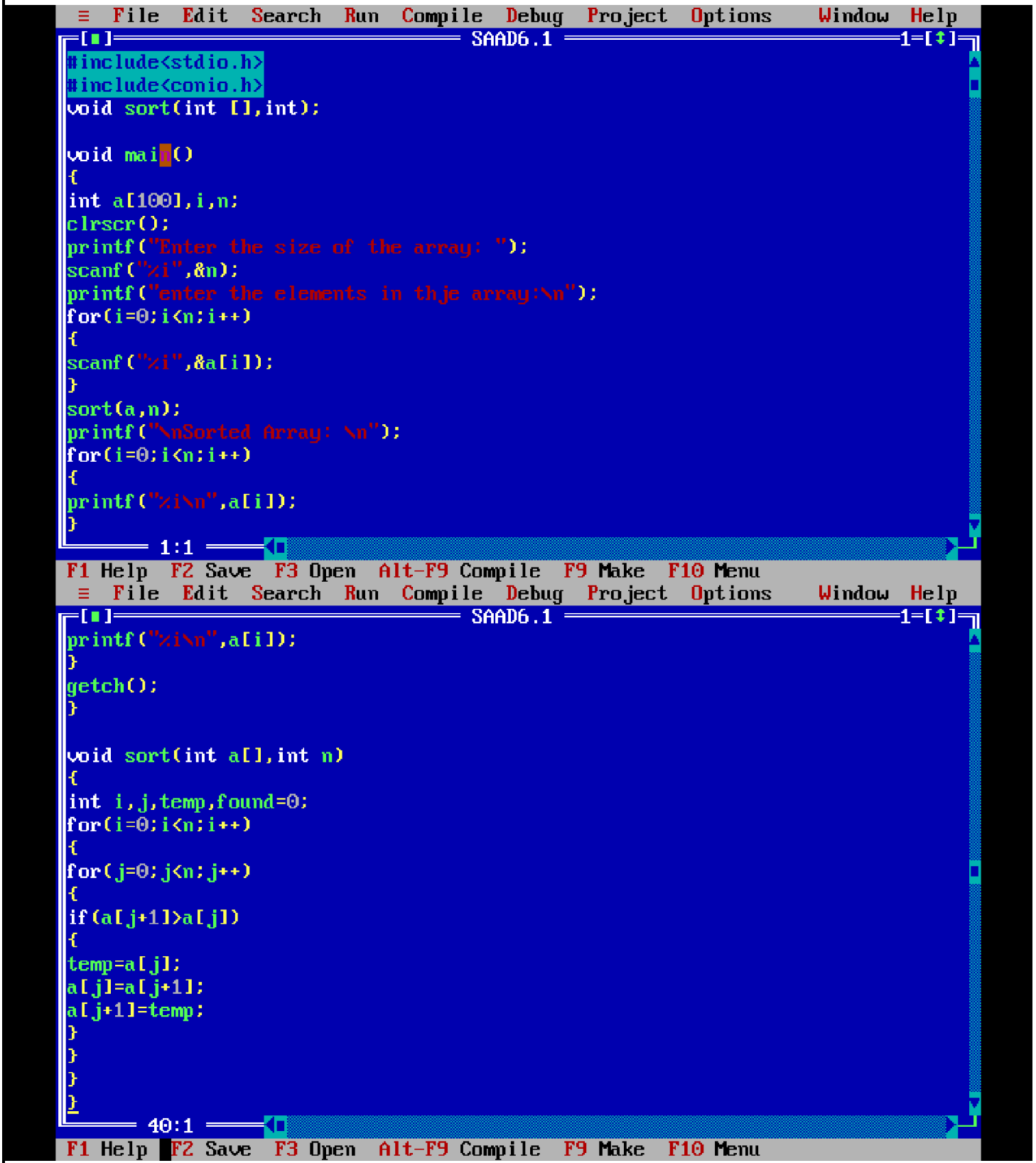
Output: -


Enter the size of the array: 5
Enter the elements in the array:
67
90
21
65
97
---Sorted Array---
21
65
67
90
97

Practical Related Questions:

1. **Optimize the Bubble Sort algorithm to stop early if the array is already sorted.**

CODE:



```
#include<stdio.h>
#include<conio.h>
void sort(int [],int);

void main()
{
int a[100],i,n;
clrscr();
printf("Enter the size of the array: ");
scanf("%i",&n);
printf("enter the elements in thje array:\n");
for(i=0;i<n;i++)
{
scanf("%i",&a[i]);
}
sort(a,n);
printf("\nSorted Array: \n");
for(i=0;i<n;i++)
{
printf("%i\n",a[i]);
}
```

```
printf("%i\n",a[i]);
}
getch();
}

void sort(int a[],int n)
{
int i,j,temp,found=0;
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
if(a[j+1]>a[j])
{
temp=a[j];
a[j]=a[j+1];
a[j+1]=temp;
}
}
}
}
```

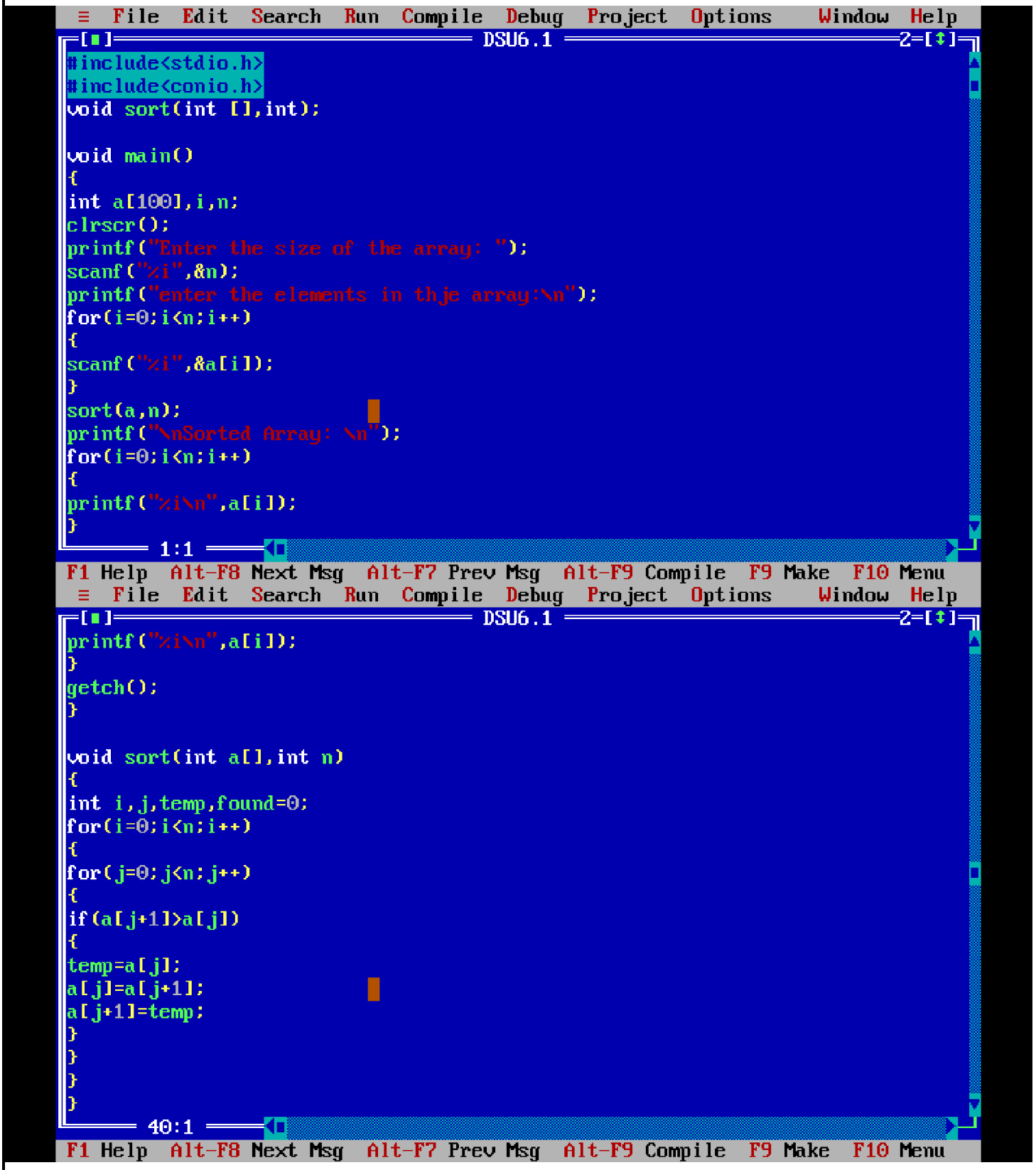
OUTPUT:

```
Enter the size of the array: 6
enter the elements in thje array:
6
5
4
3
2
1

Sorted Array:
1
2
3
4
5
6
```

1. **Modify the Bubble Sort algorithm to sort an array in descending order.**

CODE:



```
≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU6.1 2=[↕]
#include<stdio.h>
#include<conio.h>
void sort(int [],int);

void main()
{
int a[100],i,n;
clrscr();
printf("Enter the size of the array: ");
scanf("%i",&n);
printf("enter the elements in thjc array:\n");
for(i=0;i<n;i++)
{
scanf("%i",&a[i]);
}
sort(a,n);
printf("\nSorted Array: \n");
for(i=0;i<n;i++)
{
printf("%i\n",a[i]);
}
1:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
≡ File Edit Search Run Compile Debug Project Options Window Help
[■] DSU6.1 2=[↕]
printf("%i\n",a[i]);
}
getch();
}

void sort(int a[],int n)
{
int i,j,temp,found=0;
for(i=0;i<n;i++)
{
for(j=0;j<n;j++)
{
if(a[j+1]>a[j])
{
temp=a[j];
a[j]=a[j+1];
a[j+1]=temp;
}
}
}
}
40:1
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
```

OUTPUT:

```
Enter the size of the array: 5
enter the elements in thje array:
1
2
3
4
5

Sorted Array:
5
4
3
2
1
```

Marks Obtained			Dated signature of Teacher
Process Related (35)	**Product Related(15)**	**Total (50)**	