VP Vidyalankar Polytechnic	DEPARTMENT OF COMPUTER ENGINEERING

Subject: - DSU	**Subject Code:** 313301
Semester: - III	**Course:** Computer Engineering
Laboratory No: L003	**Name of Subject Teacher:** Prof. Imran S.
Name of Student: Mohd Saad Khan	**Roll Id: -** 24203A0007
Experiment No:	8
Title of Experiment	* Write a 'C' Program to Sort an Array of numbers using Selection Sort Method.

Aim: Write a 'C' Program to Sort an Array of numbers using Selection 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 the 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 integer variables i, j, min, temp
Step 9.2: Run a loop from i = 0 to i < n - 1
Step 9.2.1: Set min = i
Step 9.2.2: Run a nested loop from j = i + 1 to j < n
Step 9.2.2.1: If a[j] < a[min], then set min = j
Step 9.2.3: Swap a[i] with a[min] 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 a[i]
Step 12: Stop

Code:

```
≡ File Edit Search Run Compile Debug Project Options Window Help
[■]═══ SAAD57.C ═══1=[↕]
#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 the 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]);
}
═══ 21:78 ═══
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
[■]═══ SAAD57.C ═══1=[↕]
getch();
}

void sort(int a[],int n)
{
int i,j,min,temp=0;
for(i=0;i<n-1;i++)
{
min=i;
for(j=i+1;j<n;j++)
{
if(a[j]<a[min])
{
min=j;
}
}
temp=a[i];
a[i]=a[min];
a[min]=temp;
}
}
═══ 42:78 ═══
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
```

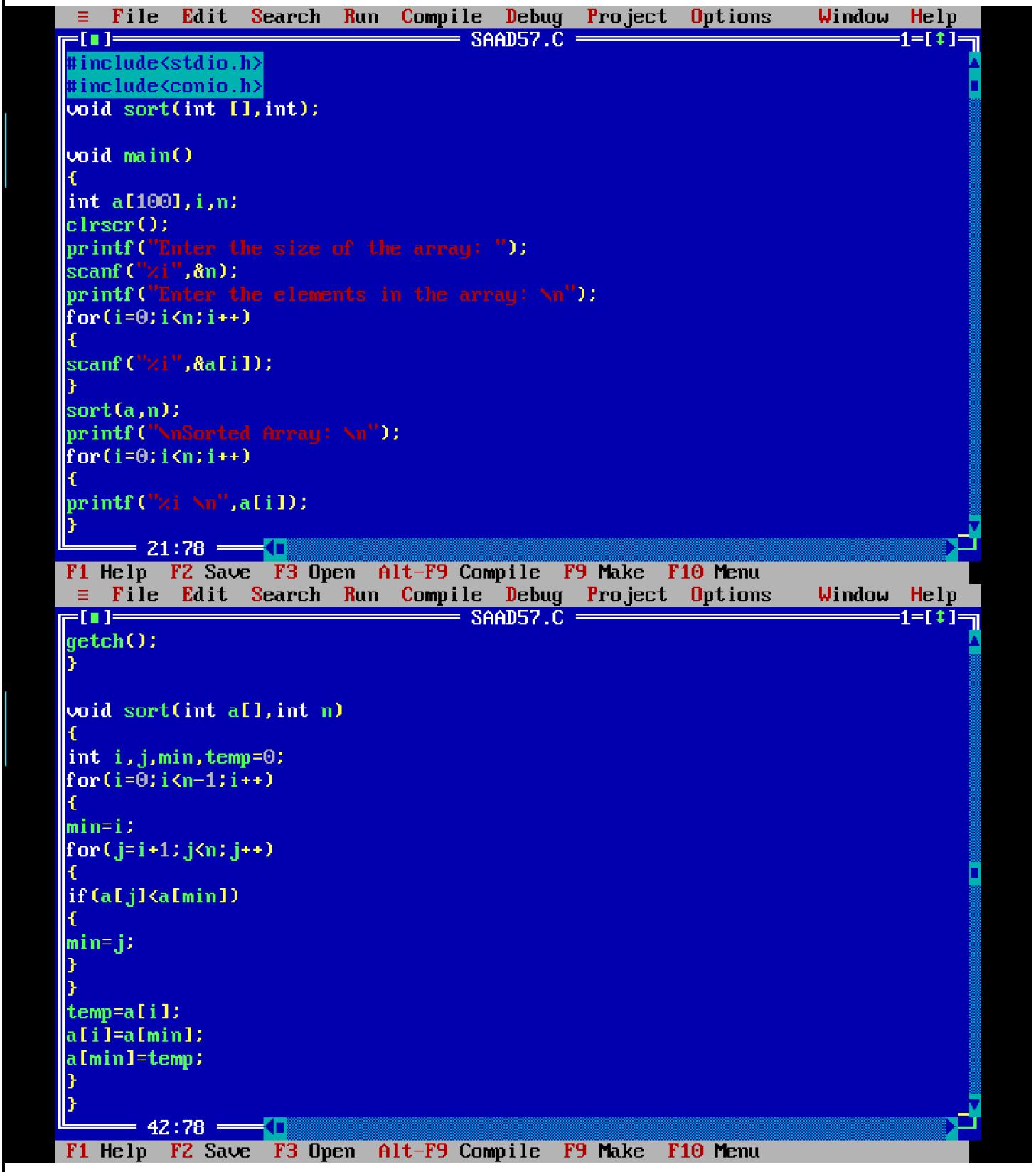
Output: -

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

Sorted Array:
1
2
3
4
5
```

Practical Related Questions:

1. **Modify the Selection Sort algorithm to handle arrays containing negative numbers.**
   Ans:



Output:

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

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

2. **Adapt the Selection Sort algorithm to sort an array of floating-point numbers.**
   Ans:

```
≡ File  Edit  Search  Run  Compile  Debug  Project  Options    Window  Help
[■]═══════════ SAAD57.C ═══════════1=[↕]
#include<stdio.h>
#include<conio.h>
void sort(float [],int);

void main()
{
float a[100];
int i,n;
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("%f",&a[i]);
}
sort(a,n);
printf("\nSorted Array: \n");
for(i=0;i<n;i++)
{
printf("%.2f \n",a[i]);
═══ 21:78 ═══
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
[■]═══════════════ SAAD57.C ═══════════════1=[↕]
}
getch();
}

void sort(float a[],int n)
{
int i,j,min;
float temp=0;
for(i=0;i<n-1;i++)
{
min=i;
for(j=i+1;j<n;j++)
{
if(a[j]<a[min])
{
min=j;
}
}
temp=a[i];
a[i]=a[min];
a[min]=temp;
═══ 42:78 ═══
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
[■]═══════════════ SAAD57.C ═══════════════1=[↕]
}
}
```

Output:

```
Enter the size of the array: 5
Enter the elements in the array:
1.5
1.4
1.3
1.2
1.1

Sorted Array:
1.10
1.20
1.30
1.40
1.50
_
```

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