

| **TITLE:**  Program to sort array |
| --- |

**AIM:** Write a C Program to sort the 1D array in the ascending or descending order and then accept the element from user and insert in the same array at its correct place by keeping array sorted. **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected OUTCOME of Experiment:**

**CO3:** Illustrate the use of derived and structured data types such as arrays, strings, structures and unions.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

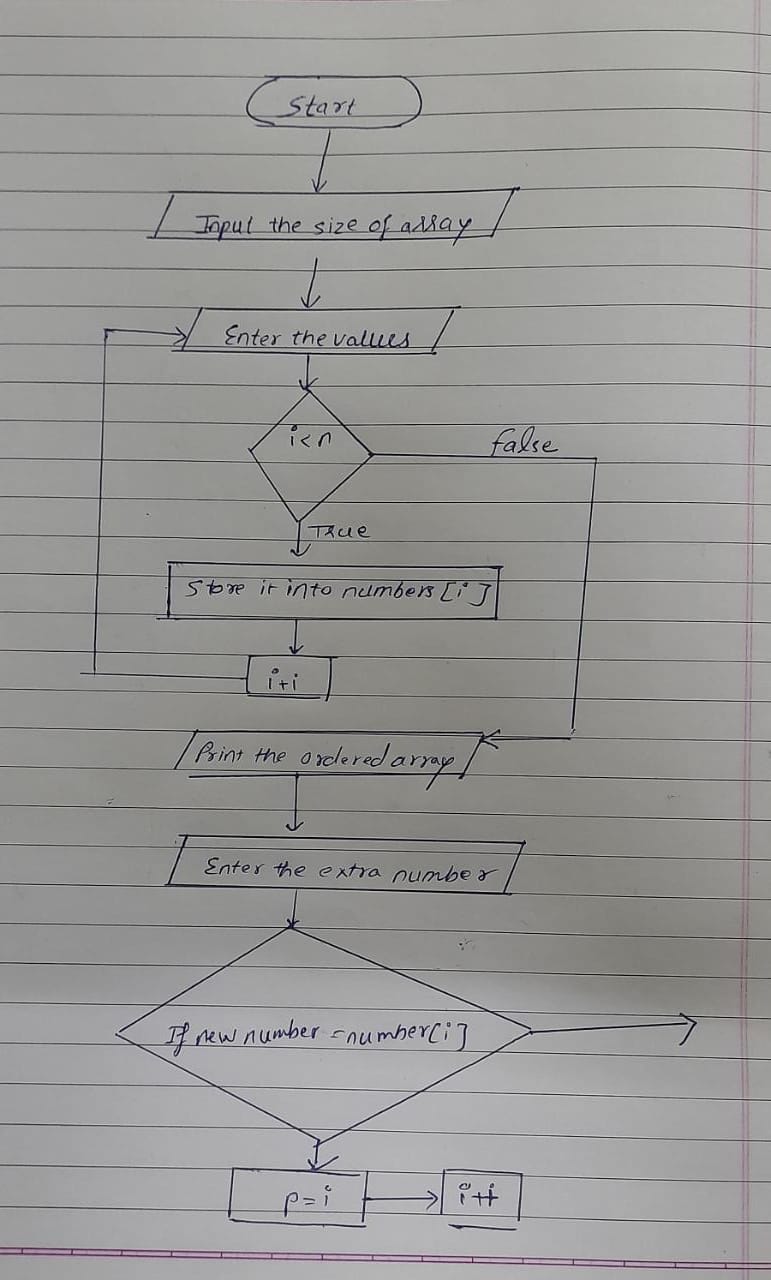
1. Programming in C, second edition, Pradeep Dey and Manas Ghosh, Oxford University Press.
2. Programming in ANSI C, fifth edition, E Balagurusamy, Tata McGraw Hill.
3. Introduction to programming and problem solving, G. Michael Schneider ,Wiley India edition.
4. [**http://cse.iitkgp.ac.in/~rkumar/pds-vlab/**](http://cse.iitkgp.ac.in/~rkumar/pds-vlab/)

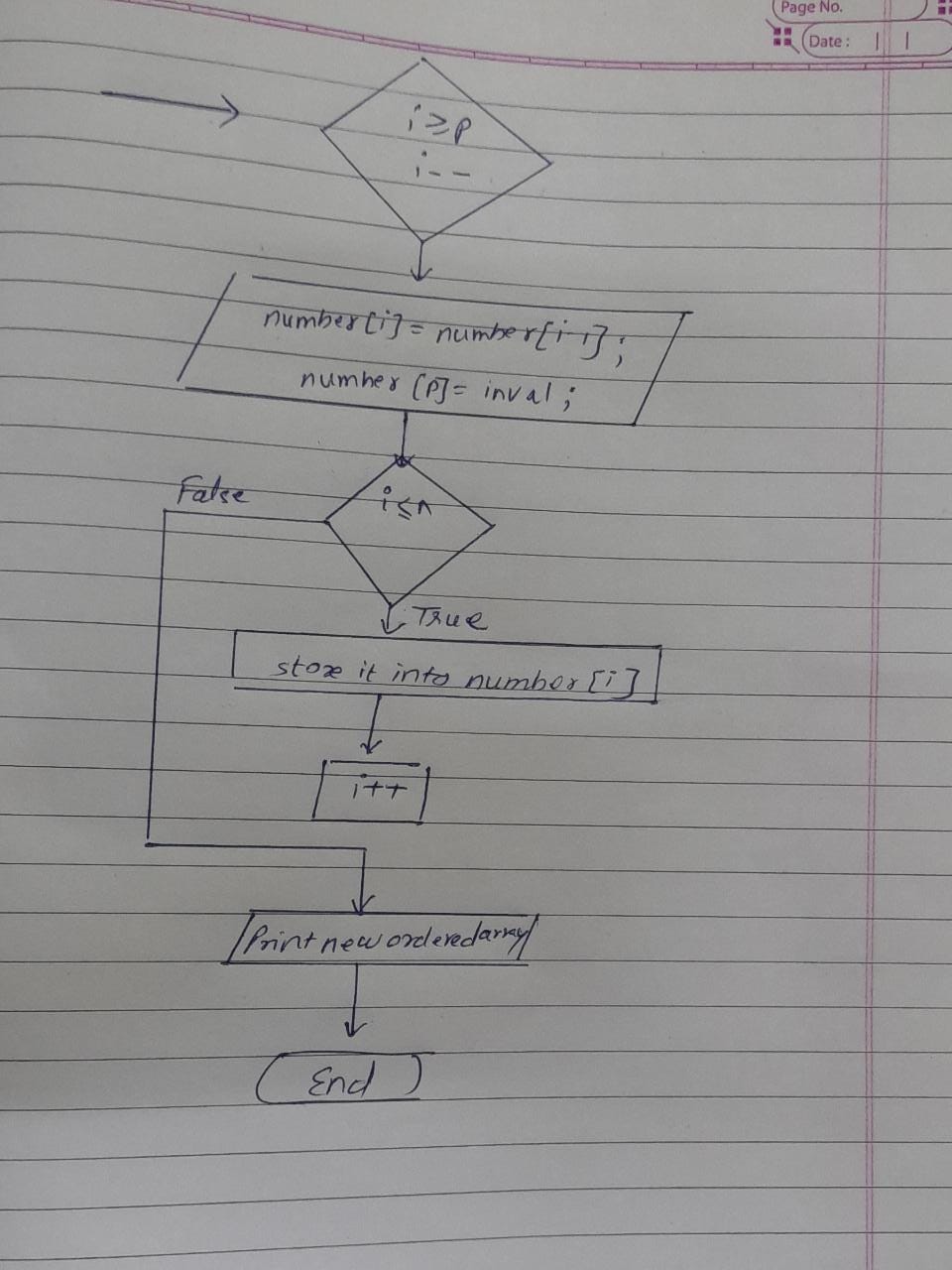
**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Problem Definition:**

The program takes a 1D array and sorts it in the specified manner. The user enters an element and the same has to be inserted at the correct place in the sorted array.

**Flowchart:**

****

****

**Implementation details:**

**// Experiment 5 Meet Gala 16010121051\_A3 //**

**#include <stdio.h>**

**#include <stdlib.h>**

**int main()**

**{**

**int i, j, a, n, number[50],inval,p;**

**printf("Enter the value of N \n");**

**scanf("%d", &n);**

**printf("Enter the numbers \n");**

**for (i = 0; i < n; i++)**

**scanf("%d", &number[i]);**

**for (i = 0; i < n; i++)**

**{**

**for (j = i + 1; j < n; j++)**

**{**

**if (number[i] > number[j])**

**{**

**a = number[i];**

**number[i] = number[j];**

**number[j] = a;**

**}**

**}**

**}**

**for(i=0;i<n;i++)**

**printf("%d\t",number[i]);**

**printf("\nInput the value to be inserted : ");**

**scanf("%d",&inval);**

**for(i=0;i<n;i++)**

**{**

**if(inval<number[i])**

**{**

**p = i;**

**break;**

**}**

**else**

**{**

**p = i+1;**

**}**

**}**

**for(i=n +1;i>=p;i--)**

**number[i]= number[i-1];**

**number[p]=inval;**

**printf("After Insert the list is :\n ");**

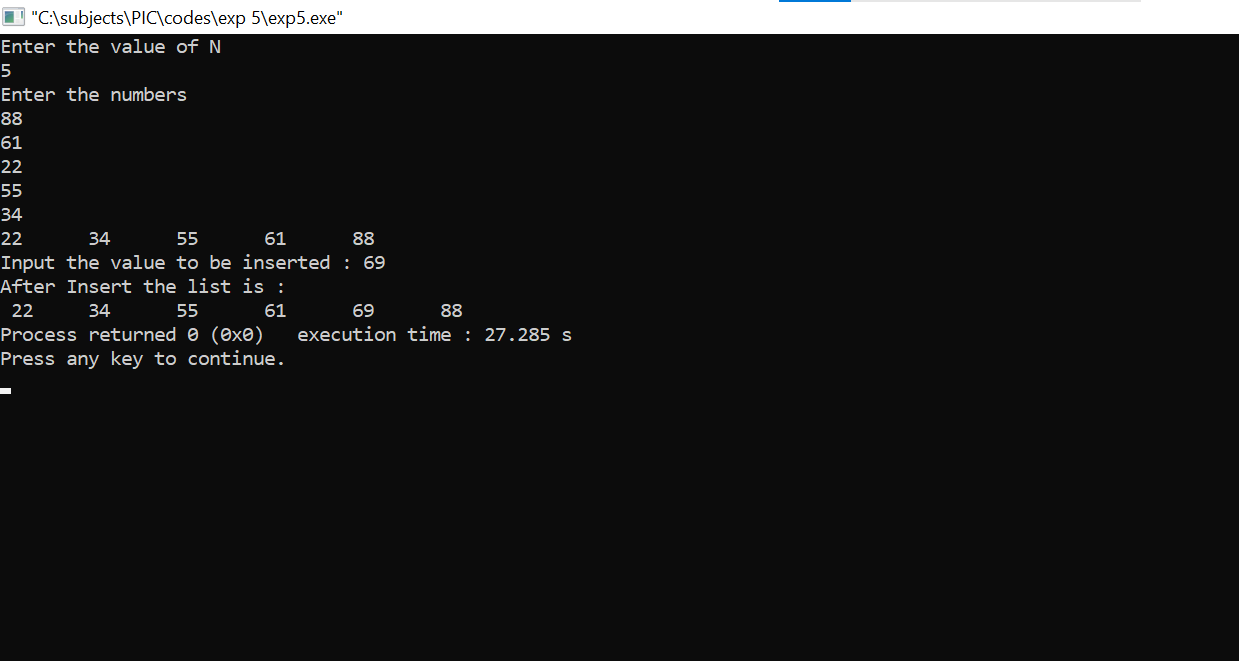
**for(i=0;i<=n;i++)**

**printf("%d\t",number[i]);**

**return 0;**

**}**

**Output(s):**



**Conclusion: The desirable 1D array program was successfully made and implemented.**

**Post Lab Descriptive Questions**

Write a program to enter n numbers, store them in an array and rearrange array in the reverse order.

Implementation :

// Experiment 5 Postlab 16010121051\_A3 //

#include <stdio.h>

int main() {

int arr[100], i, n, temp;

printf("\n\n Enter the size of array: \n");

scanf("%d",&n);

printf("\n Enter %d Elements: \n", n);

for(i=0;i<=n-1;i++)

{

scanf("%d",&arr[i]);

}

printf("Original Array Elements: \n");

for(i=0;i<=n-1;i++)

{

printf("%d \t",arr[i]);

}

for(i=0;i<=(n-1)/2;i++)

{

temp=arr[i];

arr[i]=arr[n-1-i];

arr[n-1-i]=temp;

}

printf("\n\n Reverse array is \n");

for(i=0;i<=n-1;i++)

{

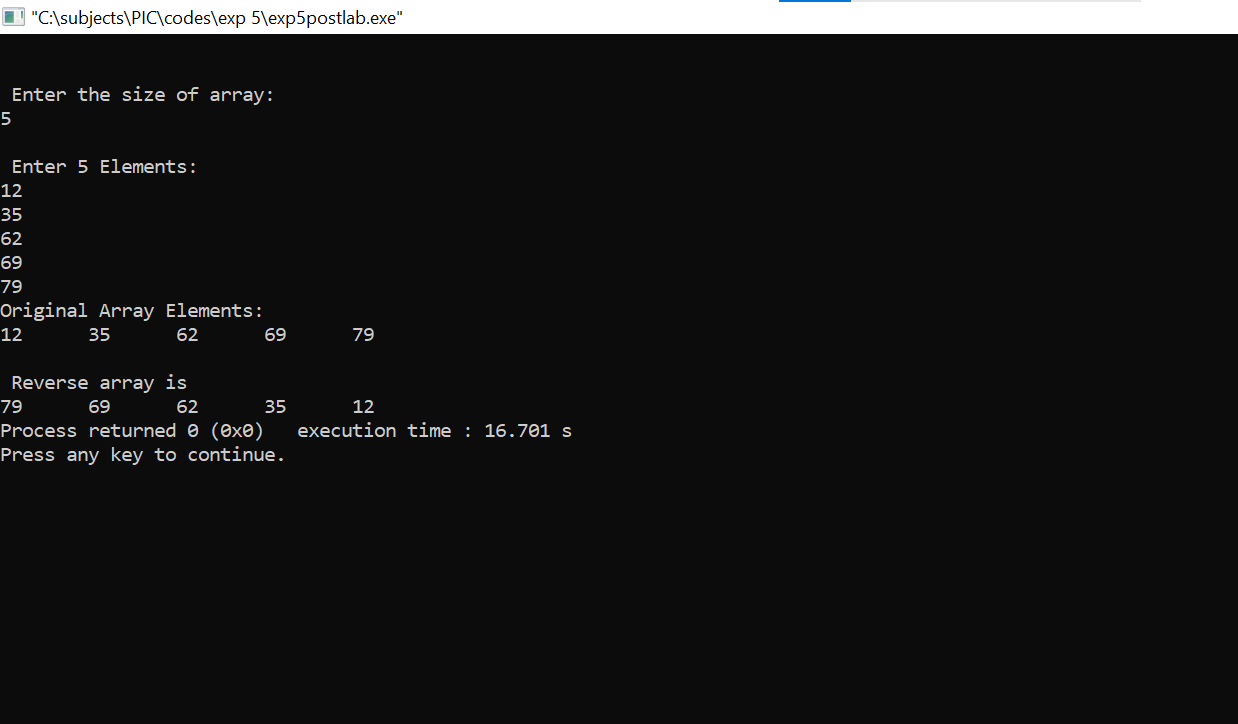
printf("%d \t",arr[i]);

}

return 0;

}

Output :-



**Date: \_\_\_\_\_\_\_\_\_\_\_\_\_ Signature of faculty in-charge**