**Batch: A2 Roll No.: 16010121045**

**Experiment / assignment / tutorial No. 5**

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of the Staff In-charge with date**

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

**AIM:** 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 datatypes such as arrays, strings, structures and unions.

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

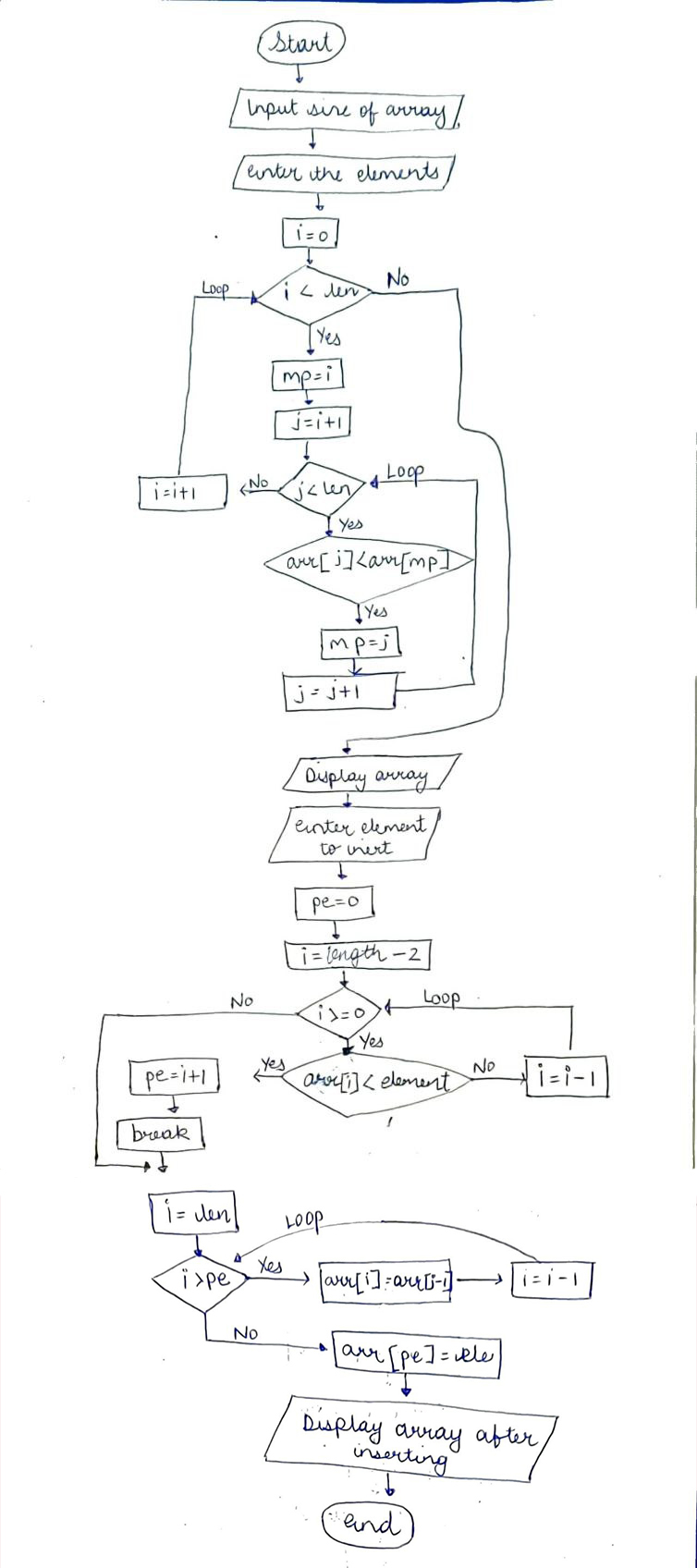
**Books/ Journals/ Websites referred:**

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:**

#include<stdio.h>

int main()

{

printf("Enter the number of elements:\n");

int len;

scanf("%d",&len);

int arr[++len];

printf("Enter the list:\n");

for(int i=0;i<len-1;i++)

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

arr[len-1]=100000;

// sorting elements

for(int i=0;i<len;i++)

{

int mp=i;

for(int j=i+1;j<len;j++)

{

if(arr[j]<arr[mp])

mp=j;

}

int temp=arr[i];

arr[i]=arr[mp];

arr[mp]=temp;

}

printf("\nData After sorting: \n");

for(int i=0;i<len-1;i++)

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

printf("\n");

int ele;

printf("Enter the element you want to insert:\n");

scanf("%d",&ele);

//finding where to insert

int pe=0;

for(int i=len-2;i>=0;i--)

{

if(arr[i]<ele){

pe=i+1;

break;}

}

//inserting here

for(int i=len;i>pe;i--)

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

arr[pe]=ele;

//completed inserting

printf("List after inserting and sorting:\n");

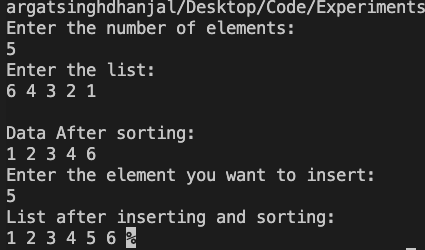
for(int i=0;i<len;i++)

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

return 0;

}

**Output(s):**



**Conclusion:**

Exp. 5 has been successfully completed 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 details:**

#include<stdio.h>

int main() {

int n;

printf("Enter the number of Elements: ");

scanf("%d",&n);

printf("Enter the elements\n");

int arr[n];

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

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

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

{

int temp=arr[i];

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

arr[n-1-i]=temp;

}

printf("In Reverse order: \n");

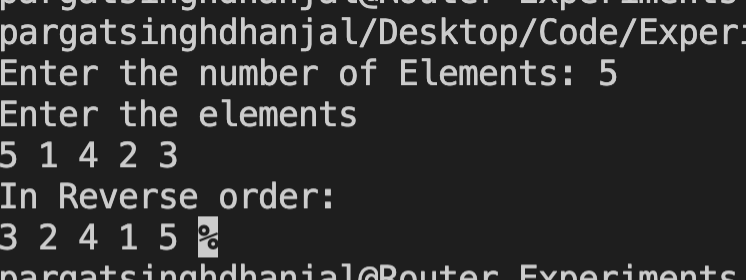
for(int i=0;i<n;i++)

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

return 0;

}

**Output**

****

**Date: 16/12/21 Signature of faculty in-charge**