Assignment – 5

The objective of this assignment is to learn how to use array in the program

1 – D array

Assignment:

1. Write a C program to find the sum of all elements of an array. Also find average.
2. Write a C program to search the any elements from an array using linear search technique.
3. Write a C program to find maximum and minimum element in an array (without sorting).
4. Write a C program to check the frequency of a given element in an array.
5. Write a C program to find the second largest element from an array (without sorting).
6. Write a C program to find the second smallest element from an array (without sorting).

7. Write a C program to sort the all elements of an array in ascending order using bubble sort technique.
8. Write a C program to sort the all elements of an array in ascending order using selection sort technique.

Practice:

1. Write a C program to calculate the sum of all negative and positive elements in an array.
2. Write a C program to find the arithmetic mean, variance and standard deviation of any n values.

   Mean $(\mu) = \frac{1}{n}\sum_{i=1}^{n} x_i$ Varianc$(\vartheta) = (\mu) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^{2}$

   Standard deviation$(\sigma) = \sqrt{\vartheta}$
3. Write a C program to search the any elements from an array using binary search technique.
4. Write a C program to sort the all elements of an array in ascending order using modified bubble sort technique.
5. Write a C program to sort the all elements of an array in ascending order using insertion sort technique.
6. Write a C program to delete all duplicate elements from an array.
7. Write a C program to left rotate and right rotate an array.

Q1. Write a C program to find the sum of all elements of an array. Also find average.

Algo:

```
// Here a is an array
//INPUT
        Print "Enter number of elements in array"
        Input n
        for(index=0; index<n; index++)
        begin
                Print "Enter data for a[index]:"
                Input  a[index])
        end

        //sum calculation
        sum=0
        for(index=0;index<n;index++)
        {
                sum=sum+a[index]
        }
        average=(float)sum / n    // Type conversion is needed here
        //Output
        Print sum, average
```

Q2. Write a C program to search the any elements from an array using linear search technique.

Algo:

```
//Input

Input number of elements n and array a

Print "Enter element/item to be searched:"

Input item

Indexpos=-1;

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

begin

//compare the item with each element

        if(a[i]==item)

        {

                Printf “\nItem”, item, “Found at position” , i+1

                Indexpos=i;

        }


end of for


//Item Not present

if(Indexpos==-1)
          print “Item  not present in array"
```

Q3. Write a C program to find maximum and minimum element in an array (without sorting).

Algo:

<u>Algo to find out minimum</u>

```
    Input number of elements n and array a
    min=a[0]
    minPos=0 // Index position of minimum
    for(i=1;i<n;i++)
    begin

        //Update minimum
        if(a[i]<min){
            min=a[i]
            minPos=i
        }
    End of for

    //Output
    Print min , minPos        // Print minimum and Index position of minimum
```

## Try finding out maximum by yourself ##

Q4. Write a C program to check the frequency of a given element in an array.

Algo:

```
// Input
Input number of elements n and array a
Print "Enter the element to be count:"
Input item

//Frequency count
count=0
for(i=0;i<n;i++)
begin
//compare the item with each element
    if(a[i]==item){
        count = count + 1
    }
End of for

if(count==0) //Item Not present
    print "\nItem  not present in array”
else
    print "\nFrequency of item is ",  count
```

```
}
```

Q5. Write a C program to find the second largest and 2nd smallest element from an array (without sorting).

```
    // max1 stores 1st maximum, max2 stores 2nd maximum

    // min1 stores 1st minimum, min2 stores 2nd minimum


    // Input

    Input number of elements n and array a


    //Initialization

    if(a[0]>a[1]){

        max1=min2=a[0]

        max2=min1=a[1]

    }

    else{

        max1=min2=a[1]

        max2=min1=a[0]

    }


// Finding out Largest, 2nd largest and smallest , 2nd smallest

    for(i=2;i<n;i++)

    begin

        //Largest and 2nd Largest

        if(a[i]>max1){

            max2=max1

            max1=a[i]

        }
```

```
            else if(a[i] > max2 && a[i] < max1)
                  max2 = a[i]



             //Smallest and 2nd smallest
             if(a[i]<min1){
                  min2=min1
                  min1=a[i]
            }
            else if(a[i] < min2 && a[i] > min1)
            {
                  min2 = a[i]
            }

End of for

//OUTPUT
Print max1,max2
Print min1,min2
```

Q6. Write a C program to sort the all elements of an array in ascending order using bubble sort technique.

Algo:

```
for(i=0;i<n-1;i++)
begin
    for(j=0;j<n-(i+1);j++)
   begin
      if(a[j]>a[j+1])
     {
   //swap a[j] with a[j+1]
        temp=a[j]
        a[j]=a[j+1]
        a[j+1]=temp
      }
    End of for
 End of for


//Output
Print "After Ascending sort: "
   for(i=0;i<n;i++)
          print a[i]
```

Q7. Write a C program to sort the all elements of an array in ascending order using selection sort technique.

```
   Algo

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

Begin

         minIndex=i

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

         Begin

               //finding index of minimum

               if(a[minIndex] > a[j])

                     minIndex=j

         End of for


         //swap a[i] with a[minIndex]

         t=a[i]

         a[i]=a[minIndex]

         a[minIndex]=t

   End of for

//Output

   Print "After Ascending sort: "

   for(i=0;i<n;i++)
          print a[i]
```