DSA LAB - 01

FARHAN JAFFER,
2105714, CSE-01

22/07/23

Q/. WAP to find out the smallest & largest element stored in an array of n integers.

A/.

```
# include <stdio.h>

int main()
{
  int size;
  printf ("Enter size of array");
  Scanf ("%d", & size);
  int arr [size];

  for (int i=0; i<size, i++)
  {
    printf ("\n Enter Number %d = ", i+1);
    scanf ("%d", & arr [i]);
  }
  int min = arr [0];
  int max = arr [0];

for (int i=0 ; i< size ; i++)
  {
    if (arr [i] >= max)
    {
    max = arr [i];
    }
    if (arr [i] <= min)
    { min = arr [i];
    }
  }
      printf ("\n Smallest %d", min);
      printf ("\n Largest %d ", max);
}
```

Output:-

Enter size of an array = 5

Enter Number 1 = 2
Enter Number 2 = 4
Enter Number 3 = 78
Enter Number 4 = 6
Enter Number 5 = 9

Smallest 2
Largest 78

- x - x - x - x - x - x -

Q2. WAP to reverse the contents of an array.

```
# include <stdio.h>

int main()
{
    int size;
    printf("Enter size of an Array ");
    scanf("%d", &size);
    int arr[size];

    for(int i=0; i<size; i++)
{
    printf("Enter number %d = ", i+1);
    scanf("%d", &arr[i]);
}
    int revarr[size];
```

```
for (int i=0; i<size; i++)
{ revarr[i] = arr[(size-1)-i];
}
for (int i=0; i<size, i++)
{
  arr[i] = revarr[i];
}
for (int i=0; i<size, i++)
{
  printf ("%d", arr[i]);
}

  return 0;
}
```

Output :-

```
Enter Size of an Array 5
Enter Number 1 = 2
Enter Number 2 = 4
Enter Number 3 = 5
Enter Number 4 = 6
Enter Number 5 = 7

  76542
```

- x - x - x - x - x -

Q3. WAP to search an element in an array of n numbers.

⇒

```
#include <stdio.h>

int main()
{
  int nbr, i, n, arr[30];

  printf("Enter the number of elements
          in the array: ");
  scanf("%d", &nbr);

  printf("Enter the array elements: ");
  for (i=0; i<nbr; i++)
  {
    scanf("%d", &arr[i]);
  }
  printf("Enter the item to be search
          -ed: ");
  scanf("%d", &n);

  i=0;
  while (i<nbr && n != arr[i])
  {
    i++;
  }

  if (i<nbr)
  {
    printf("The element is found
            in the position=%d", i+1);
  }
```

```
else {
    printf("Element not found!");
}
    return 0;
}
```

Output :-

Enter the number of elements in the array : 5

Enter the array elements : 3
5
65
6
6

Enter the item to be searched : 5
The element is found in the position = 2.

— X — X — X — X — X —

Q4. WAP to sort an array of n numbers.

Q4.
```
#include<stdio.h>
void main()
{
    int i, j, a, n, number[30];
    printf("Enter the value of N\n");
    scanf("%d", &n);
```

next page →

```
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;
        }

    }

}
    printf ("The numbers arranged in
        ascending order are given below
                \n");
    for (i=0; i<n; ++i)
        printf ("%d \n", number[i]);
}
```

OUTPUT:-

The numbers arranged in ascending order ~~are as follows~~ are given below:

next page →

OUTPUT :-

Enter the value of N
3
Enter the numbers
23
43
12
The numbers arranged in ascending order are given below
12
23
43.

— X — X — X — X — X — X —

Q5. Given an unsorted array of size n, WAP to find number of elements between two elements a & b (both inclusive).

→
```
#include <stdio.h>

int main()
{
    int size, a, b, indexa=-1, indexb=-1,
    find=0;
    printf("Enter size of Array");
     scanf("%d", &size);
    printf("Enter a");
     scanf("%d", &a);
    printf("Enter b");
     scanf("%d", &b);
```

next page

```
    int arr[size];

    for (int i=0; i<size; i++)
{
    printf ("Enter number %od = "; i+1);
     scanf ("%od", &arr[i]);
}
    for (int i=0; i<size ; i++)
{
    if (arr[i] == a && fnda==0)
    {
        indena=i;
         fnda ;
    }
     if (arr[i]==b)
    {
     indenb=i;
    }
}
    if (indena==-1 || indenb==-1)
{
    printf ("output 0");
}
    else{
    printf ("output %d\n", (indenb-
                            indena)+1);
    for (int i=indena ; i<=indenb; i++)
{
    printf ("%od", arr[i]);
}
}
    return 0;
}
```