FEW PRACTICE PROBLEMS (RELATED TO ARRAYS)

1. Write down a C++ function that searches an element in an array passed to it. The function should return the index of the first occurrence of that element, and if the data value is not present, the function should return -1. e.g., consider the following array:

Array Index	0	1	2	3	4	5	6	7	8	9
Array Data	**5**	**8**	**3**	**12**	**9**	**12**	**7**	**2**	**12**	**8**

The function call `position = FindPosition (a, 10, 12);` will update the variable `position` with 3 (because the first occurrence of data value 12 is placed at index 3).

Similarly the function call `position = FindPosition (a, 10, 8);` will update the variable `position` with 1 (because the first occurrence of data value 8 is placed at index 1).

The function call `position = FindPosition (a, 10, 4);` will update the variable position with -1 (because the data value 4 is not present in the array).

2. Write a function that receives an array and its size (through its parameters) and sorts the array in ascending order. Test this function by writing down the driver program.

3. Write a function that receives an integer array and its size (through its parameters) and uses reference parameters to return the frequency of non-zero positive integers, negative integers and zeros present in the arary.

4. Write down a C++ function definition that finds the length and location of the first longest contiguous sequence of equal values in an integer array passed to this function as input parameter. You are not allowed to use any additional array in the function.

For example, the first longest contiguous sequence is at location 8 (index 7) and its length is 4 in the following array:

1 1 1 2 2 3 2 5 5 5 5 1 5 4 4 4 4 6 4 4

A driver program for this function is as follows:

```
const int SIZE = 20;

int main()

{

        int list[20]={1,1,1,2,2,3,2,5,5,5,5,1,5,4,4,4,4,6,4,4};

        int location = 0;

        int length = 0;

        LongestSequence (list, SIZE, location, length);

        cout << “Longest contiguous sequence is at location “

        << location << “ and its length is “ << length << endl;

        return 0;

}
```

5. Write a function **ReverseString()** that reverses a C-string (an array of char) passed to it as argument.
e.g., if the string = “Computer Programming”, it should update the string to “gnimmargorP retupmoC”

6. Write a function that receives a C-string (character array) through its parameter and counts the number of vowels present in it. Write a driver program to test it.