Lab 08 – Arrays 1-D

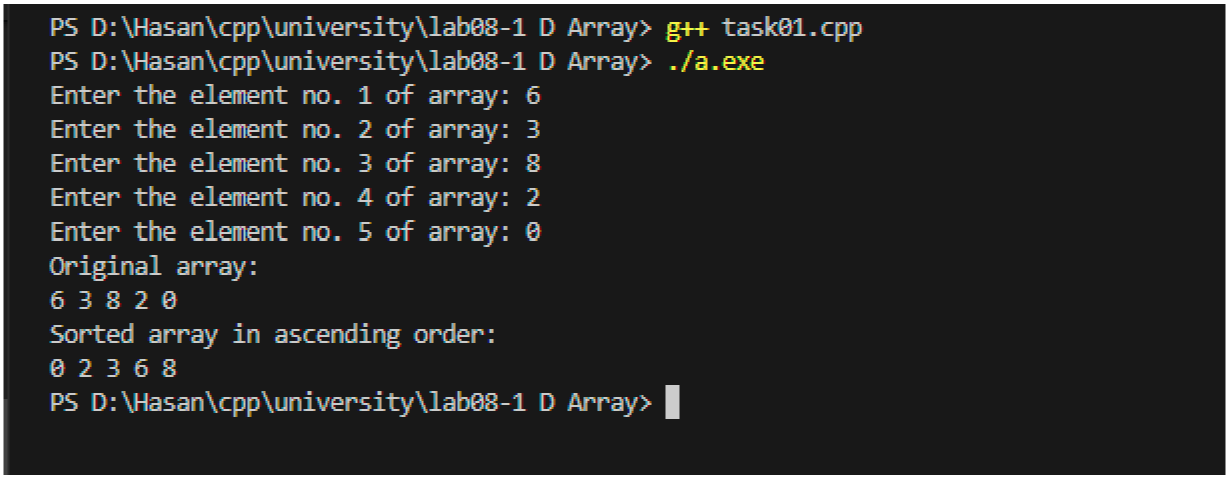
Task 01:

Write a C program to convert an array into ascending order.

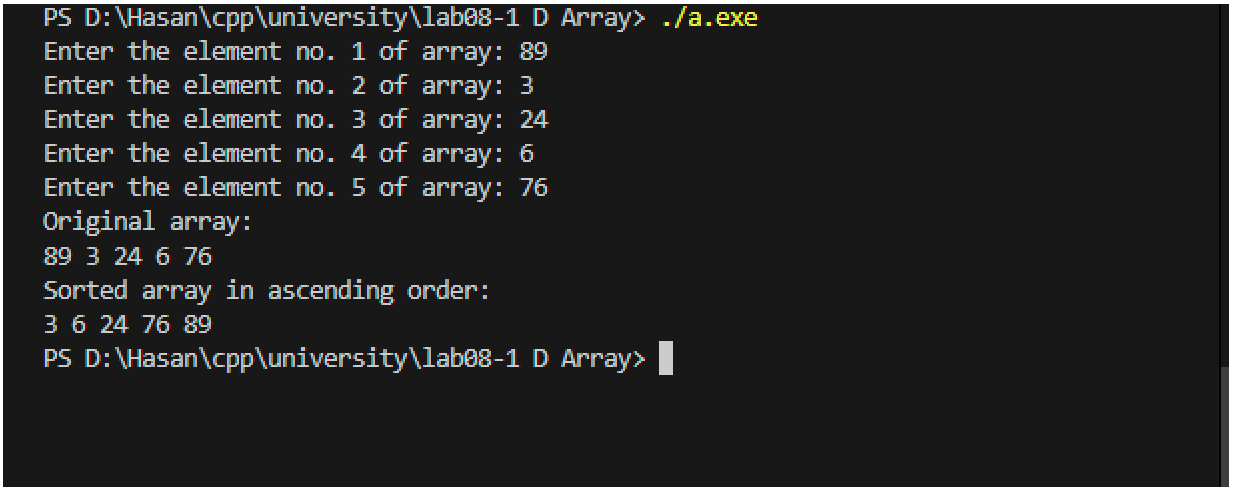
Code:

```
#include<iostream>
using namespace std;

int main(){
    int a[5];

    for(int i = 0, sNo = 1; i < 5; i++, sNo++){
        cout << "Enter the element no. "<<sNo<<" of array: ";
        cin >> a[i];
    }

    cout << "Original array:" << endl;
    for(int i = 0; i < 5; i++){
        cout << a[i] << " ";
    }

    cout << endl;
    cout << "Sorted array in ascending order:" << endl;

    //sorting
    for(int i = 0; i < (5-1); i++){
        for(int j = 0; j < 5-1-i; j++){
            if(a[j] > a[j+1]){
                int temp = a[j+1];
                a[j+1] = a[j];
                a[j] = temp;
            }
        }
    }

    //displaying sorted array
    for(int i = 0; i < 5; i++){
        cout << a[i] << " ";
    }
}
```

Output:



```
PS D:\Hasan\cpp\university\lab08-1 D Array> g++ task01.cpp
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter the element no. 1 of array: 6
Enter the element no. 2 of array: 3
Enter the element no. 3 of array: 8
Enter the element no. 4 of array: 2
Enter the element no. 5 of array: 0
Original array:
6 3 8 2 0
Sorted array in ascending order:
0 2 3 6 8
PS D:\Hasan\cpp\university\lab08-1 D Array> 
```



```
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter the element no. 1 of array: 89
Enter the element no. 2 of array: 3
Enter the element no. 3 of array: 24
Enter the element no. 4 of array: 6
Enter the element no. 5 of array: 76
Original array:
89 3 24 6 76
Sorted array in ascending order:
3 6 24 76 89
PS D:\Hasan\cpp\university\lab08-1 D Array> 
```

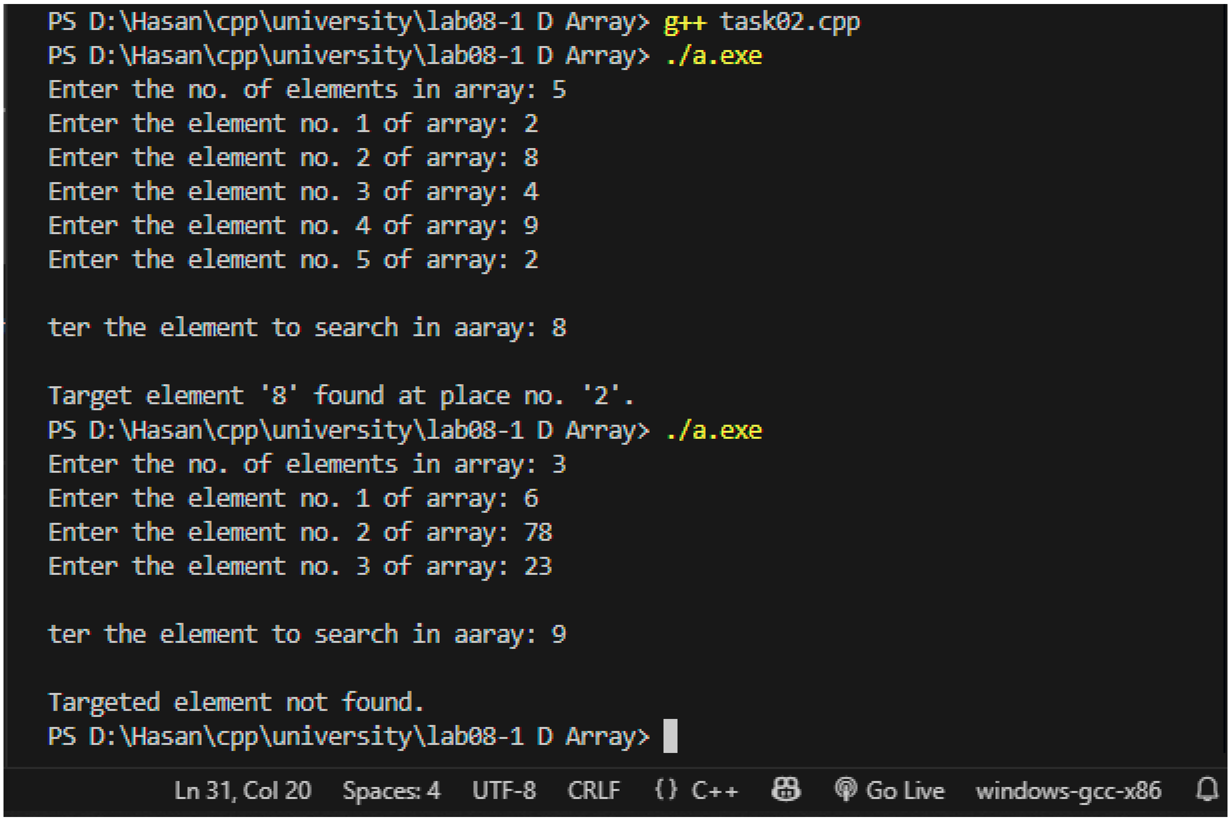
Task 02:

Write a C program to search an element entered by user from array and display the searched element and its location.

Code:

```
#include<iostream>
using namespace std;

int main(){
    int size = 0;
    int target = 0;
    bool found = false;

    cout << "Enter the no. of elements in array: ";
    cin >> size;

    int a[size];

    for(int i = 0, sNo = 1; i < size; i++, sNo++){
        cout << "Enter the element no. "<<sNo<<" of array: ";
        cin >> a[i];
    }

    cout << "Enter the element to search in aaray: ";
    cin >> target;

    for(int i = 0, sNo = 1; i < size; i++, sNo++){
        if(a[i] == target){
            found = true;
            cout << "Target element '"<<a[i]<<"' found at place no. '"<<sNo<<"'."
<< endl;
        }
    }
    if(!found){
        cout << "Targeted element not found." << endl;
    }
}
```

Output:



```
PS D:\Hasan\cpp\university\lab08-1 D Array> g++ task02.cpp
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter the no. of elements in array: 5
Enter the element no. 1 of array: 2
Enter the element no. 2 of array: 8
Enter the element no. 3 of array: 4
Enter the element no. 4 of array: 9
Enter the element no. 5 of array: 2

ter the element to search in aaray: 8

Target element '8' found at place no. '2'.
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter the no. of elements in array: 3
Enter the element no. 1 of array: 6
Enter the element no. 2 of array: 78
Enter the element no. 3 of array: 23

ter the element to search in aaray: 9

Targeted element not found.
PS D:\Hasan\cpp\university\lab08-1 D Array>
```

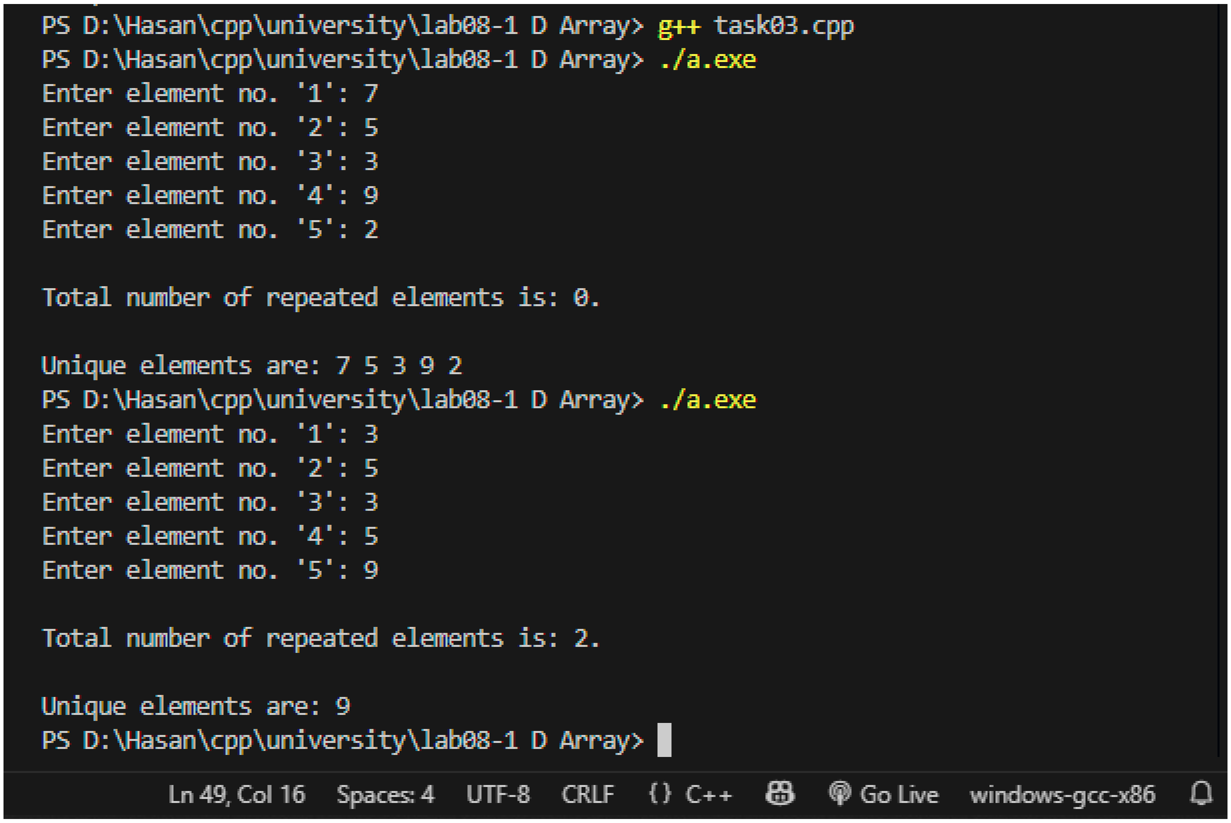
Task 03:

Write a C++ program to find total number elements repeated in an array also print all unique elements in an array.

Code:

```
#include<iostream>
using namespace std;

int main(){
    int a[5] = {0};
    int unique[5] = {0};
    int temp = 0, repeatCount = 0, uniqueCount = 0;

    //getting elements from user
    for(int i = 0; i < 5; i++){
        cout << "Enter element no. '"<<(i+1)<<"': ";
        cin >> a[i];
    }

    for(int i = 0; i < 5; i++){
        bool isRepeated = false;
        temp = a[i]; // holding the element
        for(int j = 0; j < 5; j++){
            if(i == j) continue; // not comparing element with itself
            if(temp == a[j]){
                isRepeated = true;
                break;
            }
        }

        if(isRepeated){ //checking that if the element is already counted
            bool alreadyCounted = false;
            for(int k = 0; k < i; k++){ //running the till before the holded
element
                if(temp == a[k]){
                    alreadyCounted = true;
                    break;
                }
            }
            if(!alreadyCounted){
                repeatCount++;
            }
        }
        // if it is not repeated
        if(!isRepeated){
            unique[uniqueCount] = temp;
            uniqueCount++;
        }
    }

    cout << "Total number of repeated elements is: "<<repeatCount<<"." << endl;
    cout << "Unique elements are: ";
    for(int i = 0; i < uniqueCount; i++){
        cout << unique[i] << " ";
    }
}
```

Output:



```
PS D:\Hasan\cpp\university\lab08-1 D Array> g++ task03.cpp
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter element no. '1': 7
Enter element no. '2': 5
Enter element no. '3': 3
Enter element no. '4': 9
Enter element no. '5': 2

Total number of repeated elements is: 0.

Unique elements are: 7 5 3 9 2
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter element no. '1': 3
Enter element no. '2': 5
Enter element no. '3': 3
Enter element no. '4': 5
Enter element no. '5': 9

Total number of repeated elements is: 2.

Unique elements are: 9
PS D:\Hasan\cpp\university\lab08-1 D Array>
```

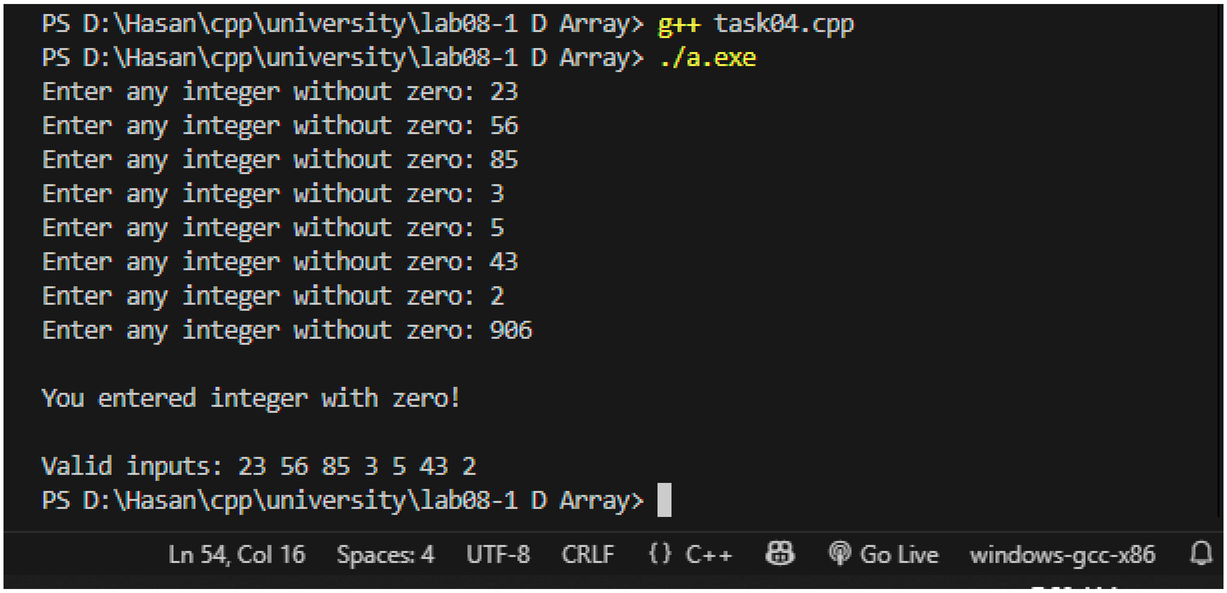
Task 04:

Write a program in C++ to identify array in which no zero present, and print those numbers. If user input a value without zero program should terminate.

Code:

```
#include<iostream>
using namespace std;


int main(){
    int max = 100;
    int validCount = 0;
    int a[max], valid[max];

    for(int i = 0; i < max; i++){
        bool zeroFound = false;
        do{
            cout << "Enter any integer without zero: ";
            cin >> a[i];

            if(a[i] == 0){
                cout << "You entered zero!" << endl;
                zeroFound = true;
                break;
            }
        }while(a[i] == 0); // if user directly enters zero. because we will not
able to check single zero with while loop

        int temp = a[i];
        int lastDigit = 0;

        while(temp != 0){
            lastDigit = temp%10; //getting the last digit
            if(lastDigit == 0){
                cout << "You entered integer with zero!" << endl;
                zeroFound = true;
                break;
            }
            else{
                temp /= 10; //removing the lastdigit if it is not equal to zero.
            }
        }

        if(zeroFound){
            break;
        }
        else{
            valid[validCount] = a[i];
            validCount++;
        }
    }

    cout << "Valid inputs: ";
    for(int i = 0; i < validCount; i++){
        cout << valid[i] << " ";
    }
}
```

Output:



```
PS D:\Hasan\cpp\university\lab08-1 D Array> g++ task04.cpp
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter any integer without zero: 23
Enter any integer without zero: 56
Enter any integer without zero: 85
Enter any integer without zero: 3
Enter any integer without zero: 5
Enter any integer without zero: 43
Enter any integer without zero: 2
Enter any integer without zero: 906

You entered integer with zero!

Valid inputs: 23 56 85 3 5 43 2
PS D:\Hasan\cpp\university\lab08-1 D Array>
```

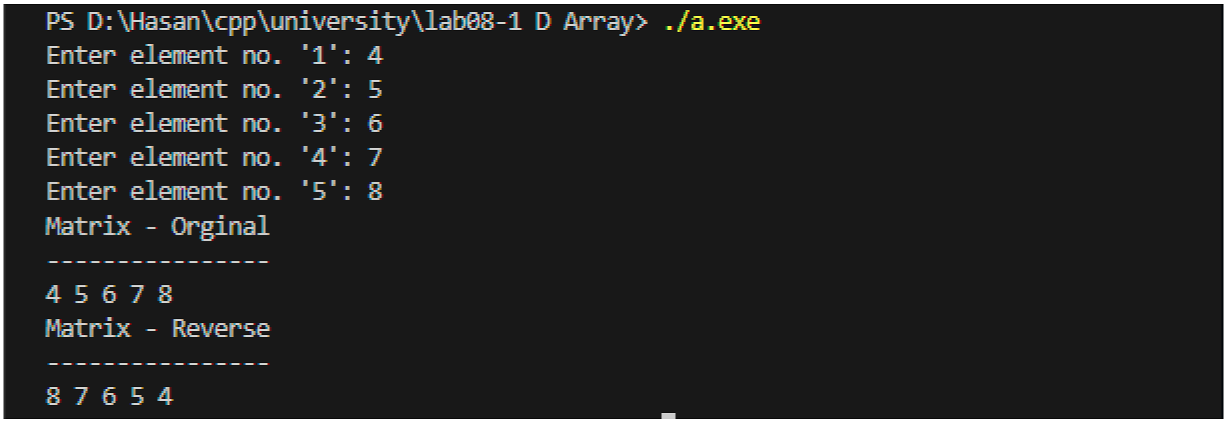
Task 05:

Write a C++ program that asks user to enter 10 integer values. Store those values in one dimensional array. Create another one dimensional array of same size, and store the values of first array in reverse order. Print the result on Screen

Code:

```cpp
#include<iostream>
using namespace std;
int main(){
    int n = 5;
    int a[n], rev[n];

    for(int i = 0; i < n; i++){
        cout << "Enter element no. '"<<(i+1)<<"': ";
        cin >> a[i];
    }
    // reversing
    for(int i = (n-1), j = 0; i >= 0; i--, j++){
        rev[j] = a[i];
    }
    cout << "Matrix - Orginal" << endl;
    cout << "----------------" << endl;
    for(int i = 0; i < n; i++){
        cout << a[i] << " ";
    }
    cout << endl << endl;
    cout << "Matrix - Reverse" << endl;
    cout << "----------------" << endl;
    for(int i = 0; i < n; i++){
        cout << rev[i] << " ";
    }
}
```

Output:



```
PS D:\Hasan\cpp\university\lab08-1 D Array> ./a.exe
Enter element no. '1': 4
Enter element no. '2': 5
Enter element no. '3': 6
Enter element no. '4': 7
Enter element no. '5': 8
Matrix - Orginal
----------------
4 5 6 7 8
Matrix - Reverse
----------------
8 7 6 5 4
```