Lab's Scope:

- **One-dimensional Arrays**

Problem 1:

Write a program to find the largest number in an array of integers. For example, if an array contains the numbers 60,80,50,20,70. The program should output that the maximum number is 80.

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 5;
    int a[SIZE] = { 60,80,50,20,70 };
    int max = a[0];

    for (int i = 1; i < SIZE; i++)
    {
        if (a[i] > max)
        {
            max = a[i];
        }
    }
    cout << "The maximum number is " << max << endl;
    return 0;
}
```

Problem 2:

Extend Problem 1 so that your program prints out the maximum number, minimum number, and the difference between them. According to the example mentioned in Problem 1, the maximum number is 80, the minimum number is 20 and the difference between them is 60.

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 5;
    int a[SIZE] = { 60,80,50,20,70 };
    int max = a[0];
    int min = a[0];
    int diff;

    for (int i = 1; i < SIZE; i++)
    {
        if (a[i] > max)
        {
            max = a[i];
        }
        if (a[i] < min)
        {
            min = a[i];
        }
    }
    diff = max - min;
    cout << "The maximum number is " << max << endl;
    cout << "The minimum number is " << min << endl;
    cout << "The difference is " << diff << endl;
    return 0;
}
```

Problem 3:

Write a program that initializes an array of even and odd integers, then prints out the odd numbers and their indexes.

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 5;
    int a[SIZE] = { 6,9,7,4,3 };
    for (int i = 0; i < SIZE; i++)
    {
        if ((a[i] % 2) != 0)
        {
            cout << "The odd number found is " <<
a[i];
            cout << " Its index is " << i << endl;
        }
    }
    return 0;
}
```

Problem 4:

Write a program that initializes two arrays of the same length, a string array to store students' names, and an integer array to store the students' ID. Assuming that the student-related information is stored at the same index in both arrays, the program should gather the information of each student from both arrays and print them out.

Example:
Given the following arrays:

Ahmed Ali	Mona Kamal	Karim Fouad	Nada Saleh

231456	238967	239564	201568

The output should be:
231456 Ahmed Ali
238967 Mona Kamal
239564 Karim Fouad
201568 Nada Saleh

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 4;
    string names[SIZE] = { "Ahmed Ali","Mona Kamal","Karim
Fouad","Nada Saleh" };
    int ids[SIZE] = { 231456,238967,239564,201568 };
    for (int i = 0; i < SIZE; i++)
    {
        cout << ids[i] << " ";
        cout << names[i] << endl;
    }
    return 0;
}
```

Problem 5:

Write a program that initializes an array of 10 integers initialized only with two numbers, the program should fill the rest of the array starting from index 2 is the addition of its previous two elements. The program should print the array elements.

Example
Given the following array:

1	2								

The output array should be:

1	2	3	5	8	13	21	34	55	89

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 10;
    int a[SIZE] = { 1,2 };

    for (int i = 2; i < SIZE; i++)
    {
        a[i] = a[i - 1] + a[i - 2];
    }
    for (int i = 0; i < SIZE; i++)
    {
        cout << a[i] << endl;
    }
    return 0;
}
```

Problem 6:

Write a program with an initialized array of El Ahly football match results where the array elements are either 1, 0, or -1. The value 1 represents a win, value 0 is a tie, whereas value -1 is a loss. The first element in the array represents match number 1, the second element represents match number 2, and so on. The program should output the match number, and whether the team wins, loses, or the match is a tie.

Example

Given the following array:

1	0	-1	1	-1

The output should be:

Match 1 El Ahly won.
Match 2 is a tie.
Match 3 El Ahly lost.
Match 4 El Ahly won.
Match 5 El Ahly lost.

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 5;
    int a[SIZE] = { 1,0,-1,1,-1 };

    for (int i = 0; i < SIZE; i++)
    {
        switch (a[i])
        {
        case 0: cout << "Match " << i + 1 << " is a tie." << endl;
                break;
        case 1: cout << "Match " << i + 1 << " El Ahly won." << endl;
                break;
        case -1:cout << "Match " << i + 1 << " El Ahly lost." << endl;
                break;
        default:cout << "Invalid number" << endl;
                break;
        }
    }
    return 0;
}
```

Problem 7:

Assume you have agreed with your friend to send each other an encrypted word, where any character is sent, your friend should replace it with the character after it in the alphabet with three letters. Write a program that initializes an array of characters with the encrypted word, the program should decrypt it, and print out the message. Manage the cases where the letters 'X', 'Y', and 'Z' in your encrypted message should be decrypted into letters 'A', 'B', and 'C' respectively.

Example:

Given the following array of characters:

T	F	K	K	B	O

The output should be:

WINNER

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 6;
    char a[SIZE] = { 'T','F','K','K','B','O' };
    int decrypt;
    for (int i = 0; i < SIZE; i++)
    {
        switch (a[i])
        {
        case 'X': decrypt = (int)'A';
            break;
        case 'Y': decrypt = (int)'B';
            break;
        case 'Z': decrypt = (int)'C';
            break;
        default:  decrypt = a[i] + 3;
            break;
        }
        cout << (char)decrypt;
    }
    return 0;
}
```

Problem 8:

Write a program that prints the count of occurrences of each element in an array of characters.

Example:

Given the following array of characters:

A	B	A	A	B	C	C	B	D	D

The output should be:

Letter A is repeated 3 times.
Letter B is repeated 3 times.
Letter A is repeated 3 times.
Letter A is repeated 3 times.
Letter B is repeated 3 times.
Letter C is repeated 2 times.
Letter C is repeated 2 times.
Letter B is repeated 3 times.
Letter D is repeated 2 times.
Letter D is repeated 2 times.

Solution:

```
#include <iostream>
using namespace std;
int main() {
    const int SIZE = 10;
    char a[SIZE] = { 'A','B','A','A','B','C','C','B','D','D' };
    int count;
    for (int i = 0; i < SIZE; i++)
    {
        count = 0;
        for (int j = 0; j < SIZE; j++)
        {
            if (a[i] == a[j])
                count++;
        }
        cout << "Letter " << a[i] << " is repeated " << count << " times." << endl;
    }

    return 0;
}
```