LAB # 01

Spring 2024

Data Structures and Algorithm

Submitted by: **Hassan Zaib Jadoon**

Registration No.: **22PWCSE2144**

Class Section: **A**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Submitted to:

Engr. Usman Mailk

February 18, 2024

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

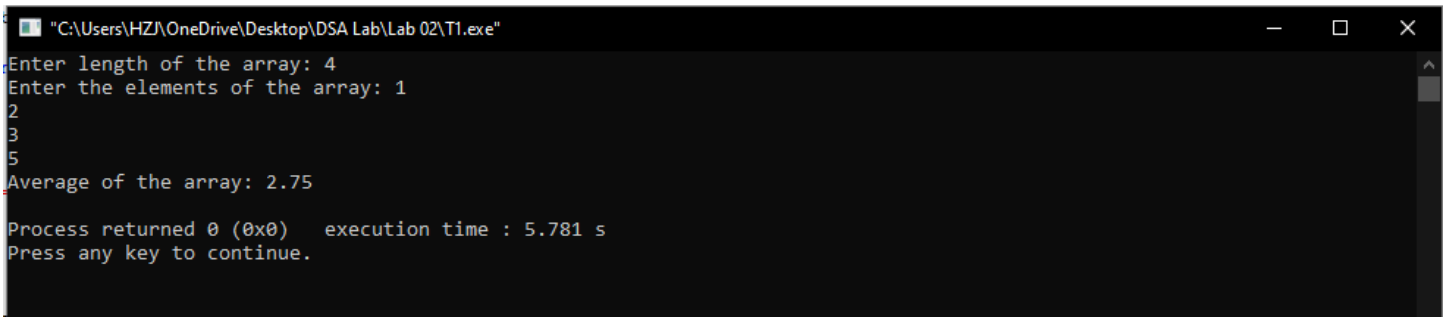
Problem 1:

Solution:

Code:

```cpp
#include <iostream>
using namespace std;
int main() {
    int length;

    cout << "Enter length of the array: ";
    cin >> length;

    float* numbers = new float[length];

    cout << "Enter the elements of the array: ";
    for (int i = 0; i < length; ++i) {
        cin >> *(numbers + i);
            }

    float sum = 0.0;
    for (int i = 0; i < length; ++i) {
        sum += *(numbers + i);
    }

    float average = sum / length;

    cout << "Average of the array: " << average << endl;
    delete[] numbers;

    return 0;
}
```

Output:



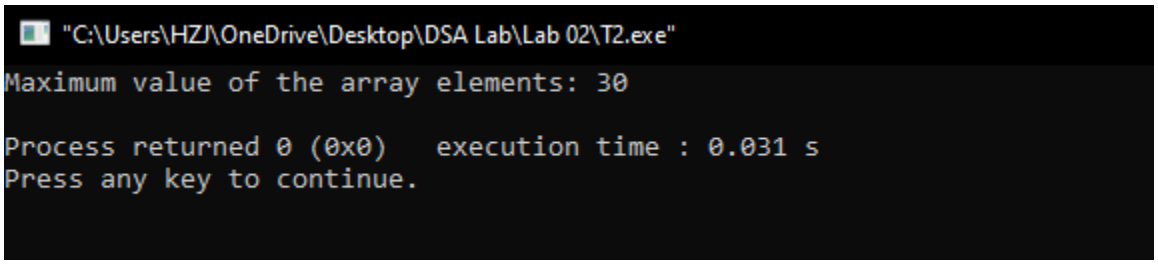
Problem 2:

Solution:

Code:

```cpp
#include <iostream>

using namespace std;

int maxValue(int arr[], int Size) {
    int temp = arr[0];

    for (int i = 1; i < Size; ++i) {
        if (arr[i] > temp) {
            temp = arr[i];
        }
    }

    return temp;
}

int main() {
    int array[] = {10, 5, 20, 15, 30};
    int size = sizeof(array) / sizeof(array[0]);

    int ReturnedValue = maxValue(array, size);
    cout << "Maximum value of the array elements: " << ReturnedValue << endl;

    return 0;
}
```

Output:



Problem 3:

Solution:

Code:

```cpp
#include <iostream>
#include <cstdlib>

using namespace std;

int* doubleSizeArray(int arr[], int size) {
    int* newArr = new int[size * 2];
    for (int i = 0; i < size; ++i) {
        newArr[i] = arr[i];
    }
    for (int i = size; i < size * 2; ++i) {
        newArr[i] = -1;
    }
    return newArr;
}

int main() {
    int arr[] = {1, 2, 3, 4, 5};
    int size = sizeof(arr) / sizeof(arr[0]);
    int* newArray = doubleSizeArray(arr, size * 2);
    cout << "New Array: ";
    for (int i = 0; i < size * 2; ++i) {
        cout << newArray[i] << " ";
    }
    cout << endl;
    delete[] newArray;

    return 0;
}
```

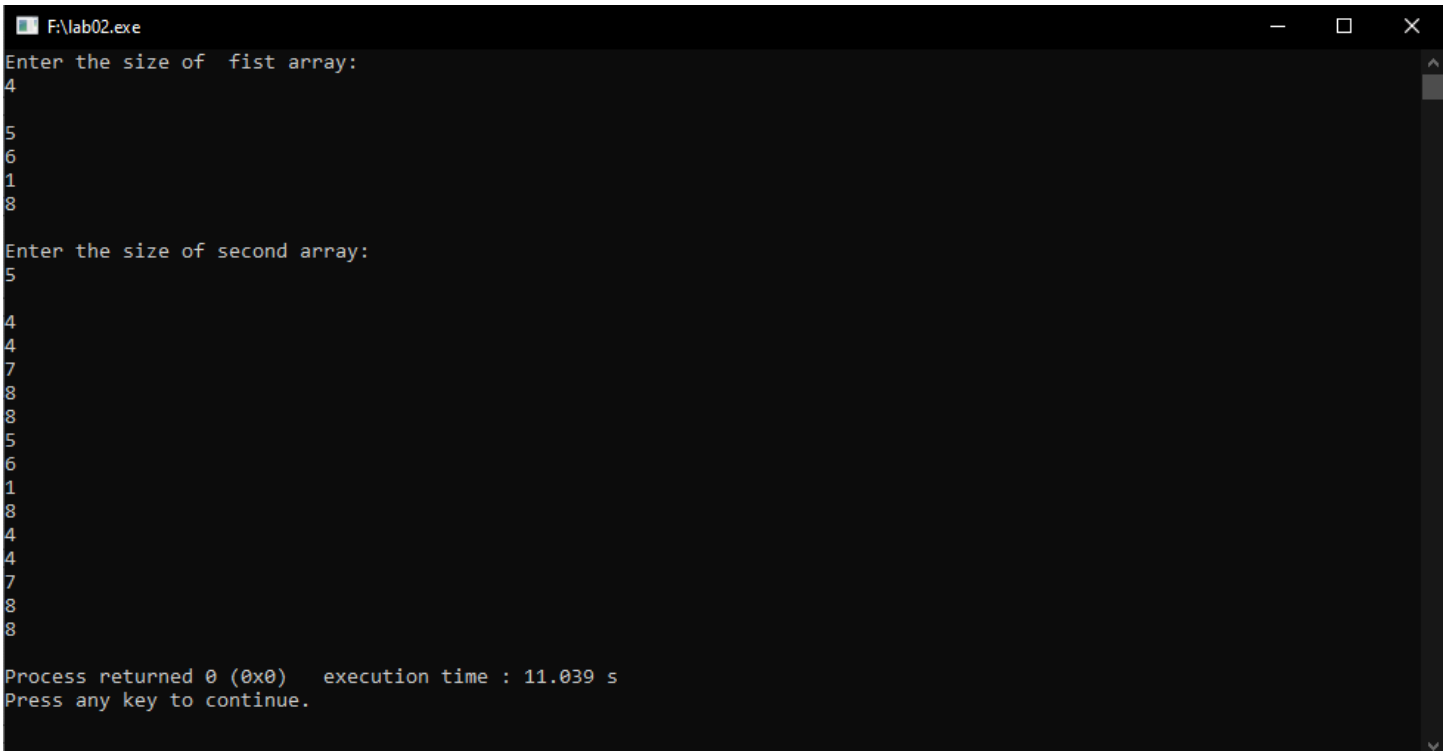
Output:

Problem 4:

Solution:
Code:

```
int mergarray(int arr1[],int arr2[],int size1,int size2)
{
    int new_size=size1+size2;
    int new_array[new_size];
    for(int i=0;i<size1;i++)
    {
        new_array[i]=arr1[i];
    }
    for(int j=0;j<size2;j++)
    {
        new_array[size1+j]=arr2[j];
    }
    for(int k=0;k<new_size;k++)
    {
        cout<<new_array[k]<<endl;
    }

}
int main()
{
    int n1,n2;
    cout<<"Enter the size of  fist array:"<<endl;
    cin>>n1;
    cout<<endl;
    int arr1[n1];
    for(int i=0;i<n1;i++)
    {
        cin>>arr1[i];
    }
    cout<<endl;
    cout<<"Enter the size of second array:"<<endl;
    cin>>n2;
    cout<<endl;
    int arr2[n2];
    for(int i=0;i<n2;i++)
    {
        cin>>arr2[i];
    }
    mergarray(arr1,arr2,n1,n2);

}
```

Output:


F:\lab02.exe
Enter the size of fist array:
4
5
6
1
8
Enter the size of second array:
5
4
4
7
8
8
5
6
1
8
4
4
7
8
8
Process returned 0 (0x0)   execution time : 11.039 s
Press any key to continue.