Name :- Ahmed Ali Asif

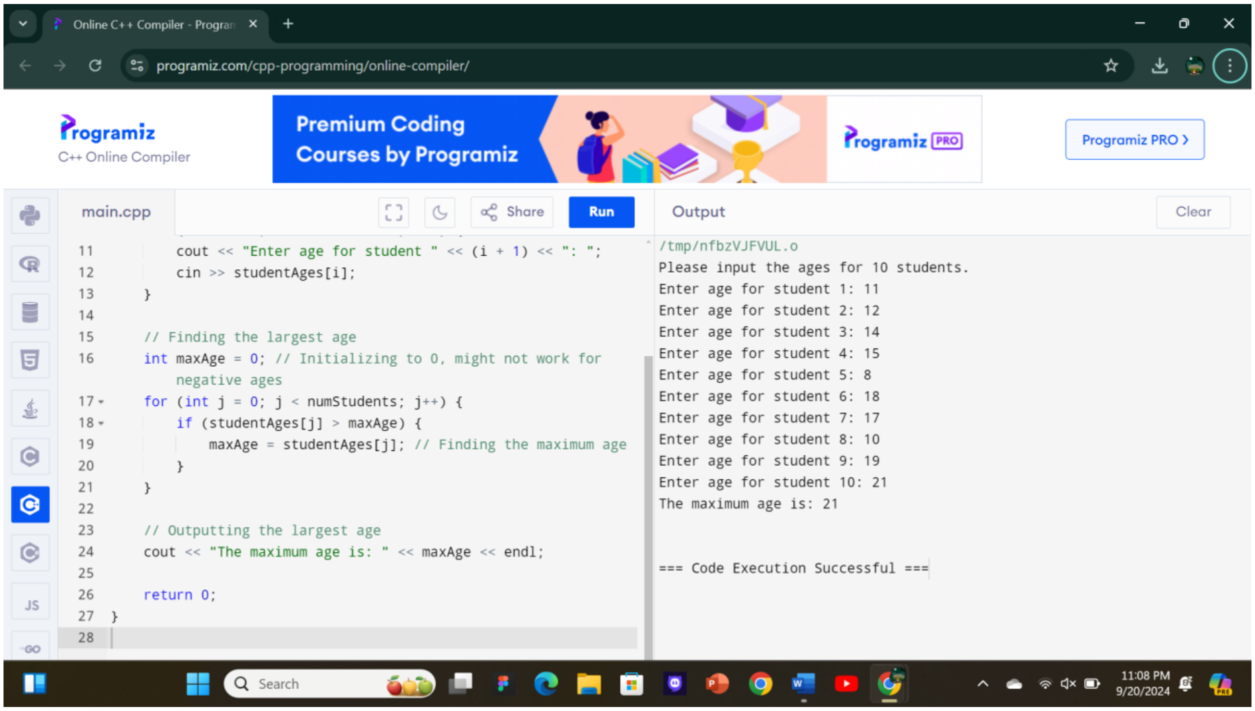
SAP ID :- 55346

Programme :- BSCS 3-1

Assignment #1

Q1

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

int main() {
    int numStudents = 10; // Not using a constant
    int studentAges[10]; // Hardcoded size
    cout << "Please input the ages for 10 students." << endl;

    // Getting ages from the user
    for (int i = 0; i < numStudents; i++) {
        cout << "Enter age for student " << (i + 1) << ": ";
        cin >> studentAges[i];
    }

    // Finding the largest age
    int maxAge = 0; // Initializing to 0, might not work for negative ages
    for (int j = 0; j < numStudents; j++) {
        if (studentAges[j] > maxAge) {
            maxAge = studentAges[j]; // Finding the maximum age
        }
    }

    // Outputting the largest age
    cout << "The maximum age is: " << maxAge << endl;

    return 0;
}
```

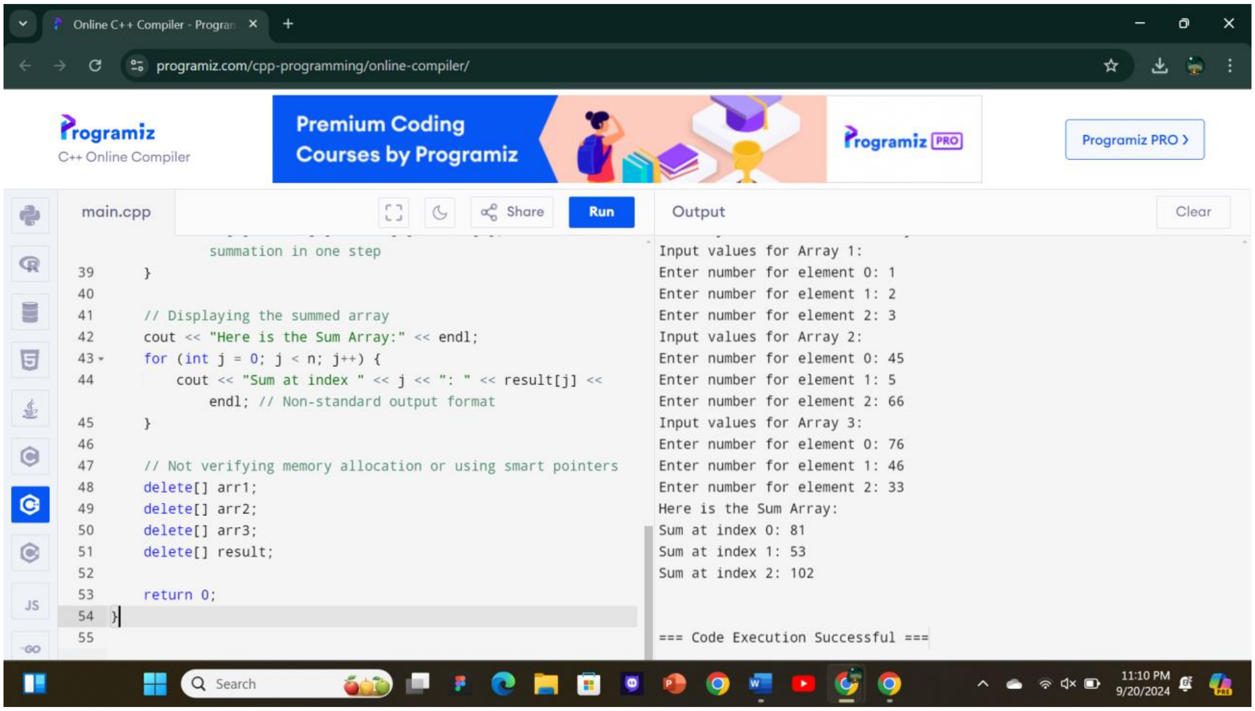


```
        cout << "Enter age for student " << (i + 1) << ": ";
        cin >> studentAges[i];
    }

    // Finding the largest age
    int maxAge = 0; // Initializing to 0, might not work for
        negative ages
    for (int j = 0; j < numStudents; j++) {
        if (studentAges[j] > maxAge) {
            maxAge = studentAges[j]; // Finding the maximum age
        }
    }

    // Outputting the largest age
    cout << "The maximum age is: " << maxAge << endl;

    return 0;
}
```

```
/tmp/nfbzVJFVUL.o
Please input the ages for 10 students.
Enter age for student 1: 11
Enter age for student 2: 12
Enter age for student 3: 14
Enter age for student 4: 15
Enter age for student 5: 8
Enter age for student 6: 18
Enter age for student 7: 17
Enter age for student 8: 10
Enter age for student 9: 19
Enter age for student 10: 21
The maximum age is: 21


=== Code Execution Successful ===
```

Q2

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

int main() {
    int n; // using a vague variable name
    cout << "How many elements for the arrays? ";
    cin >> n;

    // Using dynamic memory allocation, but could be avoided
    int *arr1 = new int[n];
    int *arr2 = new int[n];
    int *arr3 = new int[n];
    int *result = new int[n];

    // Filling array 1
    cout << "Input values for Array 1:" << endl;
    for (int idx = 0; idx < n; idx++) {
        cout << "Enter number for element " << idx << ": "; // Unnecessary repeated text
        cin >> arr1[idx];
    }

    // Filling array 2
    cout << "Input values for Array 2:" << endl;
    for (int idx = 0; idx < n; idx++) {
        cout << "Enter number for element " << idx << ": ";
        cin >> arr2[idx];
    }
    // Filling array 3
    cout << "Input values for Array 3:" << endl;
    for (int idx = 0; idx < n; idx++) {
        cout << "Enter number for element " << idx << ": ";
        cin >> arr3[idx];
    }

    // Summing the arrays
    for (int i = 0; i < n; i++) {
        result[i] = arr1[i] + arr2[i] + arr3[i]; // Direct summation in one step
    }

    // Displaying the summed array
    cout << "Here is the Sum Array:" << endl;
    for (int j = 0; j < n; j++) {
        cout << "Sum at index " << j << ": " << result[j] << endl; // Non-standard output format
    }

    // Not verifying memory allocation or using smart pointers
    delete[] arr1;
    delete[] arr2;
    delete[] arr3;
    delete[] result;
    return 0;
}
```

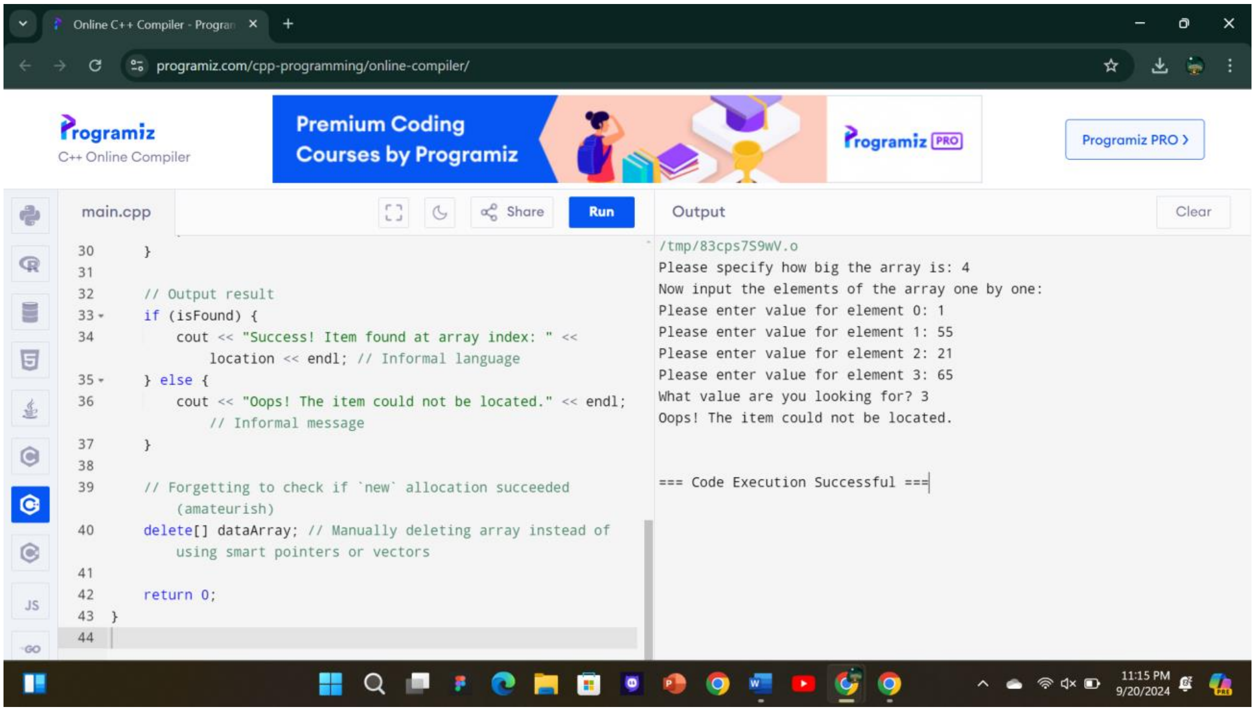

Online C++ Compiler - Progran
programiz.com/cpp-programming/online-compiler/
Programiz
C++ Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
Programiz PRO ›
main.cpp
Share
Run
Output
Clear
Search
11:10 PM
9/20/2024

```
            summation in one step
    }

    // Displaying the summed array
    cout << "Here is the Sum Array:" << endl;
    for (int j = 0; j < n; j++) {
        cout << "Sum at index " << j << ": " << result[j] <<
            endl; // Non-standard output format
    }

    // Not verifying memory allocation or using smart pointers
    delete[] arr1;
    delete[] arr2;
    delete[] arr3;
    delete[] result;

    return 0;
}
```

```
Input values for Array 1:
Enter number for element 0: 1
Enter number for element 1: 2
Enter number for element 2: 3
Input values for Array 2:
Enter number for element 0: 45
Enter number for element 1: 5
Enter number for element 2: 66
Input values for Array 3:
Enter number for element 0: 76
Enter number for element 1: 46
Enter number for element 2: 33
Here is the Sum Array:
Sum at index 0: 81
Sum at index 1: 53
Sum at index 2: 102


=== Code Execution Successful ===
```

Q3

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

int main() {
    int lengthOfArray;
    cout << "Please specify how big the array is: ";
    cin >> lengthOfArray;

    int* dataArray = new int[lengthOfArray];
    cout << "Now input the elements of the array one by one: " << endl;
    for (int index = 0; index < lengthOfArray; index++) {
        cout << "Please enter value for element " << index << ": ";
        cin >> dataArray[index];
    }

    int itemToSearch;
    cout << "What value are you looking for? ";
    cin >> itemToSearch;

    bool isFound = false;
    int location = -1;
    for (int x = 0; x < lengthOfArray; x++) {
        if (dataArray[x] == itemToSearch) {
            isFound = true;
            location = x;
            break;
        }
    }

    // Output result
    if (isFound) {
        cout << "Success! Item found at array index: " << location << endl; // Informal language
    } else {
        cout << "Oops! The item could not be located." << endl; // Informal message
    }
    delete[] dataArray; // Manually deleting array instead of using smart pointers or vectors


    return 0;
}
```



Online C++ Compiler - Progran
programiz.com/cpp-programming/online-compiler/
Programiz
C++ Online Compiler
Premium Coding Courses by Programiz
Programiz PRO
Programiz PRO ›
main.cpp
Share
Run
Output
Clear
11:15 PM
9/20/2024

```
    }

    // Output result
    if (isFound) {
        cout << "Success! Item found at array index: " <<
            location << endl; // Informal language
    } else {
        cout << "Oops! The item could not be located." << endl;
            // Informal message
    }

    // Forgetting to check if `new` allocation succeeded
        (amateurish)
    delete[] dataArray; // Manually deleting array instead of
        using smart pointers or vectors

    return 0;
}
```

```
/tmp/83cps7S9wV.o
Please specify how big the array is: 4
Now input the elements of the array one by one:
Please enter value for element 0: 1
Please enter value for element 1: 55
Please enter value for element 2: 21
Please enter value for element 3: 65
What value are you looking for? 3
Oops! The item could not be located.


=== Code Execution Successful ===
```