

Computer Systems & Programming

Instructor:
Dr. Talha
Shahid



Assignment # 1

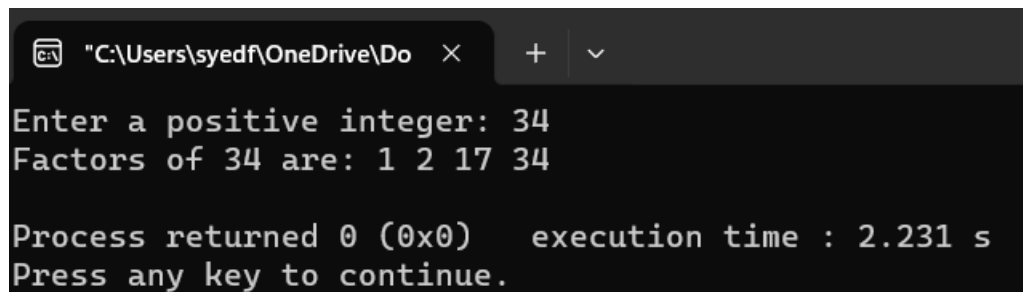
SYED FAKHAR ABBAS

ME-15-C

466960

Task#1: Write a C++ program to display factors of a number using for loops.

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int a, b;
5      cout << "Enter a number: ";
6      cin >> a;
7      cout << "The factors of " << a << " are: ";
8      for (b = 1; a <= b; ++b) {
9          if (a % b == 0) {
10             cout << b << " ";
11         }
12     }
13     cout << endl;
14     return 0;
15 }
16
```



```
"C:\Users\syedf\OneDrive\Do  x  +  v
Enter a positive integer: 34
Factors of 34 are: 1 2 17 34

Process returned 0 (0x0)   execution time : 2.231 s
Press any key to continue.
```

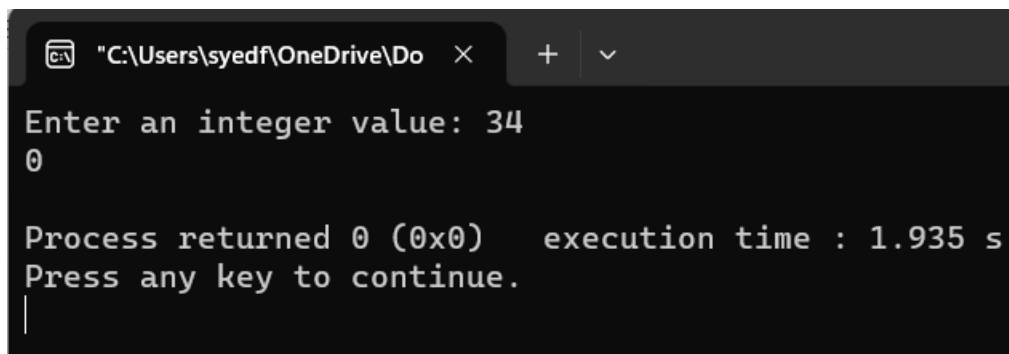
Task# 2: Write output to the following code.

```
#include <iostream>
int main() {
    int x = 5;
    int y = 10;
    if (x == 5)
    if (y == 10)
        std::cout << "x is 5 and y is 10" << std::endl;
    else
        std::cout << "x is not 5" << std::endl;
    return 0;
}
```

OUTPUT: x is 5 and y is 10

Task# 3: Write a C++ program, take an integer value from user and check if it's greater than 10 and less than equal to 20. Print 1 if yes and print 0 if no. Use appropriate datatype for output.

```
1  #include <iostream>
2  using namespace std;
3  int main() {
4      int num;
5      cout << "Enter an integer value: ";
6      cin >> num;
7
8      if (num > 10 && num <= 20) {
9          cout << "1" << endl;
10     } else {
11         cout << "0" << endl;
12     }
13     return 0;
14 }
15
```



```
"C:\Users\syedf\OneDrive\Do  x  +  v
Enter an integer value: 34
0
Process returned 0 (0x0)  execution time : 1.935 s
Press any key to continue.
|
```

Task# 4: Write a C++ program that uses a **while** loop to find the largest prime number less than a given positive integer **N**. Your program should take the value of **N** as input from the user and then find the largest prime number less than or equal to **N**. You are not allowed to use any library or pre-existing functions to check for prime numbers.

Code:

```
#include <iostream>
```

```

using namespace std;
bool check_primity(int num) {
    if (num <= 1) {
        return false;
    }

    for (int i = 2; i * i <= num; ++i) {
        if (num % i == 0) {
            return false;
        }
    }

    return true;
}

int identify_largest_prime(int limit) {
    if (limit <= 2) {
        return -1;
    }

    int largest_prime_so_far = -1;
    for (int number = limit - 1; number >= 2; --number) {
        if (check_primity(number)) {
            largest_prime_so_far = number;
            break;
        }
    }

    return largest_prime_so_far;
}

int main() {
    int upper_limit;
    cout << "Enter a positive integer value N: ";
    cin >> upper_limit;

    int largest_prime_found = identify_largest_prime(upper_limit);
    if (largest_prime_found == -1) {
        cout << "No prime number found less than N" << endl;
    } else {
        cout << "The largest prime number less than N is: " << largest_prime_found << endl;
    }

    return 0;
}

```

```
"C:\Users\syedf\OneDrive\Do  × + v
Enter a positive integer value N: 45
The largest prime number less than N is: 43

Process returned 0 (0x0)    execution time : 2.097 s
Press any key to continue.
}
```

Task# 5: Write a C++ program, take two string as input from user and check if both strings are equal or not. If they are equal make them unequal by rotating string. e.g., Hello is turned into olleH etc.

Code:

```
#include <iostream>
#include <string>

using namespace std;

int main() {
    string word1, word2;

    cout << "Enter two words: ";
    cin >> word1 >> word2;

    if (word1 == word2) {
        reverse(word1.begin(), word1.end());
        cout << "Reversed word: " << word1;
    } else {
        cout << "The words are not the same.";
    }

    return 0;
}
```

```
C:\Users\syedf\OneDrive\Do  × + v
Enter the first string: 23
Enter the second string: 44
Both strings are already unequal.

Process returned 0 (0x0)   execution time : 3.578 s
Press any key to continue.
|
```

Task# 6: Perform division in C++ **without** / using **for** loops. You can use / only to display the final results. Your dividend must be greater than divisor.

Code:

```
#include <iostream>

using namespace std;

int main() {
    int numericalValue, dividingFactor;
    cin >> numericalValue >> dividingFactor;

    if (numericalValue < dividingFactor)
        return 1;

    int result = 0;
    while (numericalValue >= dividingFactor) {
        numericalValue -= dividingFactor;
        result++;
    }

    cout << result << endl << numericalValue << endl;
    return 0;
}
```

```
"C:\Users\syedf\OneDrive\Do  × + v
Enter dividend (should be greater than divisor): 45
Enter divisor: 23
Quotient: 1
Remainder: 22

Process returned 0 (0x0)   execution time : 7.685 s
Press any key to continue.
|
```

Task# 7: Write a C++program for a string which may contain lowercase and uppercase characters. The task is to remove all duplicate characters from the string and find the resultant string.

Code:

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

int main() {
    string word;
    cout << "Enter your word: " << endl;
    cin >> word;
    int len = word.length();

    for (int a = 0; a < len; ++a) {
        for (int b = a + 1; b < len; b++) {
            if (word[a] == word[b]) {
                for (int c = b; c < len - 1; ++c) {
                    word[c] = word[c + 1];
                }
                --len;
            } else {
                ++b;
            }
        }
    }
}
```

```

    cout << "Resultant string after removing duplicates: " << word.substr(0, len) << endl;
    return 0;
}

```

```

C:\Users\syedf\OneDrive\Do >
Enter your string: Helloo Worldd
Resultant string after removing duplicates: Helo Wrd

Process returned 0 (0x0)   execution time : 3.314 s
Press any key to continue.

```

Task# 8: Suppose an integer array $a[5] = \{1,2,3,4,5\}$. Add more elements to it and display them in C++.

Code:

```
#include <iostream>
```

```

int main() {
    int a[5] = { 1, 2, 3, 4, 5 };
    int* b = new int[8];
    for (int i = 0; i < 5; i++) b[i] = a[i];
    b[5] = 6; b[6] = 7; b[7] = 8;
    for (int i = 0; i < 8; i++) std::cout << b[i] << " ";
    delete[] b;
    return 0;
}

```

```

C:\Users\syedf\OneDrive\Do >
1 2 3 4 5 6 7 8
Process returned 0 (0x0)   execution time : 0.052 s
Press any key to continue.

```


Task# 9: Given an integer array and an integer **X**. Find if there's a triplet in the array which sums up to the given integer **X**.

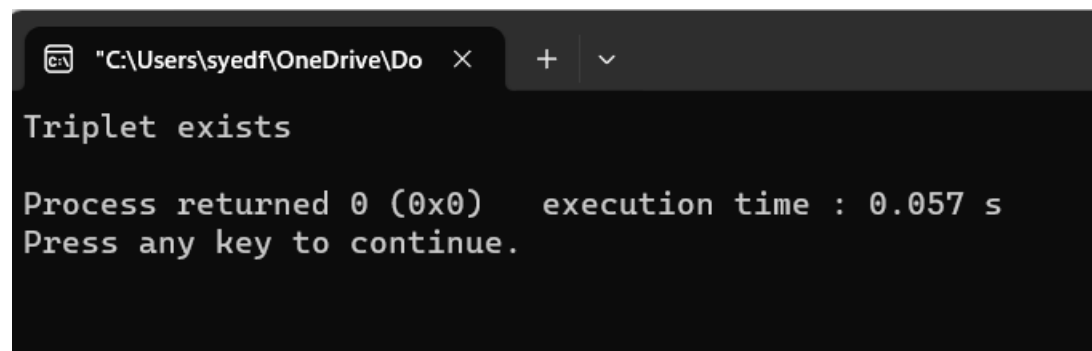
Code:

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

bool findTriplet(int arr[], int n, int X) {
    for (int i = 0; i < n - 2; i++)
        for (int j = i + 1; j < n - 1; j++)
            for (int k = j + 1; k < n; k++)
                if (arr[i] + arr[j] + arr[k] == X)
                    return true;
    return false;
}

int main() {
    int arr[] = {1, 4, 45, 6, 10, 8};
    int X = 22;
    int n = sizeof(arr) / sizeof(arr[0]);
    if (findTriplet(arr, n, X)) {
        cout << "Triplet exists\n";
    } else {
        cout << "Triplet doesn't exist\n";
    }

    return 0;
}
```



The screenshot shows a Windows command prompt window with a single tab titled "C:\Users\syedf\OneDrive\Do". The window has a dark background and displays the output of the C++ program. The first line of output is "Triplet exists". Below this, it shows "Process returned 0 (0x0)" and "execution time : 0.057 s". The final line of output is "Press any key to continue.", which is followed by a cursor.

```
"C:\Users\syedf\OneDrive\Do" × + ∨
Triplet exists
Process returned 0 (0x0)   execution time : 0.057 s
Press any key to continue.
```

Task# 10: Implement Bubble Sort on an array of 6 integers.

Code:

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

void bubbleSort(int arr[], int n) {
    for (int i = 0; i < n - 1; i++)
        for (int j = 0; j < n - i - 1; j++)
            if (arr[j] > arr[j + 1])
                swap(arr[j], arr[j + 1]);
}

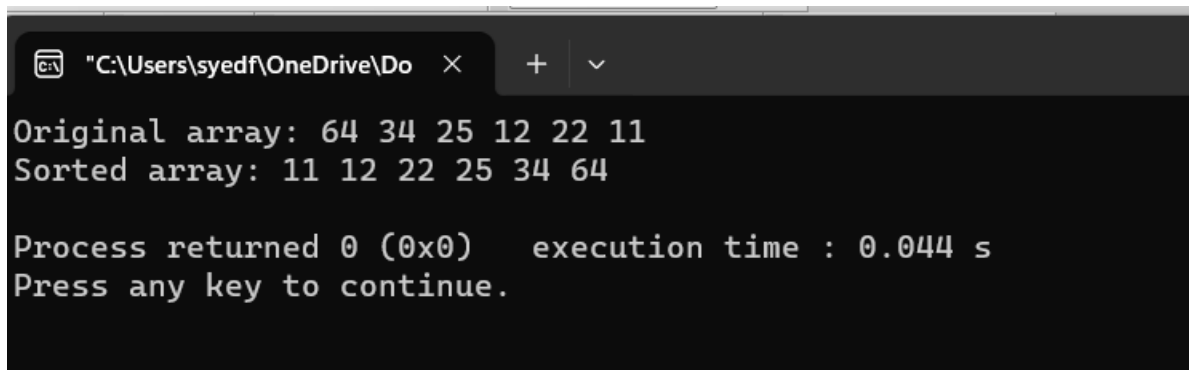
int main() {
    int arr[6] = {64, 34, 25, 12, 22, 11};
    int n = sizeof(arr) / sizeof(arr[0]);

    cout << "Original array: ";
    for (int i = 0; i < n; i++)
        cout << arr[i] << " ";
    cout << endl;

    bubbleSort(arr, n);

    cout << "Sorted array: ";
    for (int i = 0; i < n; i++)
        cout << arr[i] << " ";
    cout << endl;

    return 0;
}
```



```
"C:\Users\syedf\OneDrive\Do  ×  +  v

Original array: 64 34 25 12 22 11
Sorted array: 11 12 22 25 34 64

Process returned 0 (0x0)    execution time : 0.044 s
Press any key to continue.
```