



NUST
NATIONAL UNIVERSITY
OF SCIENCES & TECHNOLOGY

Computer Programming Lab

Assignment # 1

Submitted by:

Shah Jahan khan

469192 (section C)

Submitted to:

Sir Affan Tariq

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

```
#include<iostream>
using namespace std;
int main(){
    int num;
    cout<<" Enter a number:";
    cin>>num;
    if(num<=0){
        cout<<" Please enter a positive number!"<<endl;
        return 1;
    }
    cout<<" The factors of"<<num<<" are:";
    for(int i=1; i<num;i++){
        if(num%i==0){
            cout<<i<<" ";
        }
    }
    cout<<endl;
    return 0;
}
```

```
1  #include<iostream>
2  using namespace std;
3  int main(){
4      int num;
5      cout<<" Enter a number:";
6      cin>>num;
7      if(num<=0){
8          cout<<" Please enter a positive number!"<<endl;
9          return 1;
10     }
11     cout<<" The factors of"<<num<<" are:";
12     for(int i=1; i<num;i++){
13         if(num%i==0){
14             cout<<i<<" ";
15         }
16     }
17     cout<<endl;
18     return 0;
19 }
```

```
Output
/tmp/KqGkPX9eAx.o
Enter a number:120
The factors of120 are:1, 2, 3, 4, 5, 6, 8, 10, 12, 15, 20, 24, 30, 40, 60,
```

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;
}
```

ANSWER:

x is 5 and y is 10

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.

```
#include<iostream>
using namespace std;
int main(){
    int num;
    cout<<" Enter a number:";
    cin>>num;
    if(num>10 && num<=20){
        cout<<"1"<<endl;
    }
    else
        cout<<"0"<<endl;

    return 0;
}
```

```
1 #include<iostream>
2 using namespace std;
3 int main(){
4     int num;
5     cout<<" Enter a number:";
6     cin>>num;
7     if(num>10 && num<=20){
8         cout<<"1"<<endl;
9     }
10    else
11        cout<<"0"<<endl;
12
13    return 0;
14 }
15
16
```

[/http://SX7oc2T9Q6.o](http://SX7oc2T9Q6.o)
Enter a number:15
1

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.

```
#include<iostream>
```

```
using namespace std;
```

```
int main(){
```

```
    int divisor, dividend;
```

```
    int quotient=0, remainder=0;
```

```
    cout<<" Enter your divisor:";
```

```
    cin>>divisor;
```

```
    cout<<" Enter your dividend:";
```

```
    cin>>dividend;
```

```
    if(dividend<divisor){
```

```
        cout<<" Enter the dividend greater than or equal to divisor";
```

```
        return 1;
```

```
    }
```

```
    while(dividend>=divisor){
```

```
        dividend-= divisor;
```

```
        quotient++;
```

```
    }
```

```
    cout<<"Quotient is: "<<quotient<<endl;
```

```
    cout<<"Remainder is: "<<remainder<<endl;
```

```
    return 0;
}
```

```
1  #include<iostream>
2  using namespace std;
3  int main(){
4      int divisor, dividend;
5      int quotient=0, remainder=0;
6      cout<<" Enter your divisor:";
7      cin>>divisor;
8      cout<<" Enter your dividend:";
9      cin>>dividend;
10     if(dividend<divisor){
11         cout<<" Enter the dividend greater than or equal to divisor";
12         return 1;
13     }
14     while(dividend>=divisor){
15         dividend-= divisor;
16         quotient++;
17     }
18
19     cout<<"Quotient is: "<<quotient<<endl;
20     cout<<"Remainder is: "<<remainder<<endl;
21     return 0;
22 }
```

```
/tmp/zfqtSQJxbX.o
Enter your divisor:12
1Enter your dividend:44
Quotient is: 3
Remainder is: 0
```

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.

```

#include <iostream>

using namespace std;

bool isPrime(int num) {
    if (num <= 1) {
        return false;
    }
    for (int i = 2; i * i <= num; ++i) {
        if (num % i == 0) {
            return false;
        }
    }
    return true;
}

int main() {
    int N;

    std::cout << "Enter a positive integer N: ";
    std::cin >> N;

    int largestPrime = 0;

    while (N > 1) {
        if (isPrime(N)) {
            largestPrime = N;
            break;
        }
        --N;
    }

    if (largestPrime != 0) {

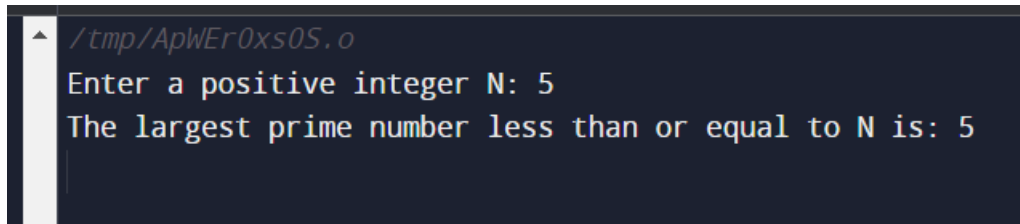
```

```

        cout << "The largest prime number less than or equal to N is: " << largestPrime << endl;
    } else {
        cout << "There is no prime number less than or equal to N." << endl;
    }

    return 0;
}

```



```

/tmp/ApWEr0xs0S.o
Enter a positive integer N: 5
The largest prime number less than or equal to N is: 5

```

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

```

#include <iostream>
#include<string>
#include<algorithm>
using namespace std;
int main() {
    string str1, str2;
    cout<<" Enter the first string:";
    cin>>str1;
    cout<<" Enter the second string: ";
    cin>>str2;
    if(str1==str2){
        rotate(str1.begin(),str1.begin()+1, str1.end());
        cout<<" Strings were equal. first string after rotation:"<<str1<<endl;
        cout<<" Second string: "<<str2<<endl;
    }
    else{

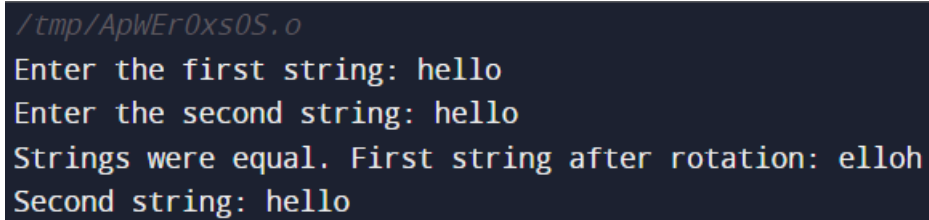
```

```

        cout<<" Strings were unequal already."<<endl;
    }

    return 0;
}

```



A terminal window with a dark background and light-colored text. The prompt is `/tmp/ApWEr0xs0S.o`. The user enters `hello` for the first string and `hello` for the second string. The program outputs `Strings were equal. First string after rotation: elloh` and `Second string: hello`.

```

/tmp/ApWEr0xs0S.o
Enter the first string: hello
Enter the second string: hello
Strings were equal. First string after rotation: elloh
Second string: hello

```

10. Implement Bubble Sort on an array of 6 integers.

```

#include <iostream>

using namespace std;

int bubbleSort(int arr[], int size) {
    for (int i = 0; i < size - 1; ++i) {
        bool swapped = false;

        for (int j = 0; j < size - i - 1; ++j) {
            if (arr[j] > arr[j + 1]) {
                // Swapping elements if they are in the wrong order
                swap(arr[j], arr[j + 1]);
                swapped = true;
            }
        }
    }

    // If no two elements were swapped in the inner loop, array is already sorted
    if (!swapped) {

```



```

        break;
    }
}

int main() {
    int arr[] = {6, 3, 8, 5, 1, 9};
    int size = sizeof(arr) / sizeof(arr[0]);

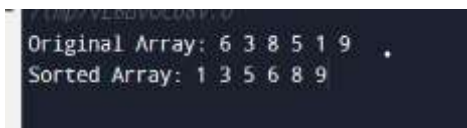
    cout << "Original Array: ";
    for (int i = 0; i < size; ++i) {
        cout << arr[i] << " ";
    }
    cout << endl;

    // Sorting the array using Bubble Sort
    bubbleSort(arr, size);

    cout << "Sorted Array: ";
    for (int i = 0; i < size; ++i) {
        cout << arr[i] << " ";
    }
    cout << endl;

    return 0;
}

```



The screenshot shows the output of the provided C++ code. It displays two lines of text: "Original Array: 6 3 8 5 1 9" followed by a space, and "Sorted Array: 1 3 5 6 8 9" followed by a space. The text is white on a dark background.

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

```

#include<iostream>

using namespace std;

int main(){

    int arr[10]={1,2,3,4,5};

    for(int i=0; i<=9; i++){

        cin>>arr[i];

    }

    for(int i=0; i<=9; i++){

        cout<<arr[i]<<endl;

    }

    return 0;

}

```

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.

```

#include <iostream>

#include <algorithm>

#include <vector>

bool findTriplet(std::vector<int>& arr, int X) {

    int n = arr.size();

    sort(arr.begin(), arr.end());

```

```

for (int i = 0; i < n - 2; ++i) {
    int left = i + 1;
    int right = n - 1;

    while (left < right) {
        int currentSum = arr[i] + arr[left] + arr[right];

        if (currentSum == X) {
            return true; // Triplet found
        } else if (currentSum < X) {
            left++; // Increment left pointer for a larger sum
        } else {
            right--; // Decrement right pointer for a smaller sum
        }
    }
}

return false; // Triplet not found
}

```

```

int main() {
    std::vector<int> arr = {1, 4, 45, 6, 10, 8};
    int X = 13;

    bool result = findTriplet(arr, X);
    if (result) {
        std::cout << "Triplet found" << std::endl;
    } else {
        std::cout << "Triplet not found" << std::endl;
    }

    return 0;
}

```

```
}
```



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

```
#include <iostream>
```

```
std::string removeDuplicates(std::string str) {
```

```
    std::string result = "";
```

```
    for (char c : str) {
```

```
        // Check if the character is already present in the result string
```

```
        if (result.find(c) == std::string::npos) {
```

```
            result += c; // If not present, add it to the result string
```

```
        }
```

```
    }
```

```
    return result;
```

```
}
```

```
int main() {
```

```
    std::string inputString = "aABbCcDdeE";
```

```
    std::cout << "Original String: " << inputString << std::endl;
```

```
    std::string resultString = removeDuplicates(inputString);
```

```
    std::cout << "Resultant String after removing duplicates: " << resultString << std::endl;
```

```
return 0;  
}
```

7tmp7nm5130cccB.0

Original String: aABbCcDdeE

Resultant String after removing duplicates: aABbCcDdeE