

# **CS-114-FUNDAMENTALS OF PROGRAMMING**

## **LAB MANUAL #6**

**COURSE INSTRUCTOR: DR TALHA SHAHID**

**LAB INSTRUCTOR: MUHAMMAD AFFAN**

**STUDENT NAME: BUSHRA FAROOQ**

**CMS ID: 479973**

**DATE:24-11-2023**



## LAB TASK 01

Generate the Fibonacci sequence using nested loops.

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

    int n=10;

    int x=0, y=1;

    cout <<x<< " " <<y<< " ";

    for (int i=2; i<n; ++i) {

        int nextTerm = x + y;

        cout << nextTerm << " ";

        x = y;
        y = nextTerm;
    }
    return 0;
}
```



```
main.cpp
1 #include <iostream>
2 using namespace std;
3 int main() {
4
5     int n=10;
6
7     int x=0, y=1;
8
9     cout <<x<< " " <<y<< " ";
10
11     for (int i=2; i<n; ++i) {
12
13         int nextTerm = x + y;
14
15
16         cout << nextTerm << " ";
17
18         x = y;
19         y = nextTerm;
20     }
21 }
```

Output

```
/tmp/xn3btGGNEe.o
0 1 1 2 3 5 8 13 21 34
```

## LAB TASK 02

Create Pascal's triangle with nested loops.

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    // Number of rows in Pascal's Triangle
```

```
    int numRows = 5;
```

```
    for (int i = 0; i < numRows; ++i) {
```

```
        int coefficient = 1;
```

```
        for (int j = 0; j < numRows - i; ++j) {
```

```
            cout << "  ";
```

```
        }
```

```
        for (int j = 0; j <= i; ++j) {
```

```
            cout << "  " << coefficient;
```

```
            coefficient = coefficient * (i - j) / (j + 1); }
        }
```

```
    cout << endl;
```



```

    }

    return 0;
}

```

```

1 #include <iostream>
2 using namespace std;
3 int main() {
4     // Number of rows in Pascal's Triangle
5     int numRows = 5;
6
7     for (int i = 0; i < numRows; ++i) {
8         int coefficient = 1;
9         for (int j = 0; j < numRows - i; ++j) {
10             cout << " ";
11         }
12         for (int j = 0; j <= i; ++j) {
13             cout << " " << coefficient;
14
15             coefficient = coefficient * (i - j) / (j + 1);
16         }
17         cout << endl;
18     }
19     return 0;
}

```

/tmp/flCQdt3tXw.o

```

1
1
1 2 1
1 3 3 1
1 4 6 4 1

```

## HOME TASK 01

Write a program using break or continue statement that only adds prime numbers from 1 to 50 and display the sum on screen.

```

#include <iostream>

using namespace std;

int main() {

    int sum = 0;

    for (int num = 2; num <= 50; ++num) {

        bool isPrime = true;

        for (int i = 2; i <= num / 2; ++i) {

            if (num % i == 0) {

                isPrime = false;

                break;
            }
        }

        if (isPrime) {
            sum += num;
        }
    }

    cout << sum << endl;
}

```



```

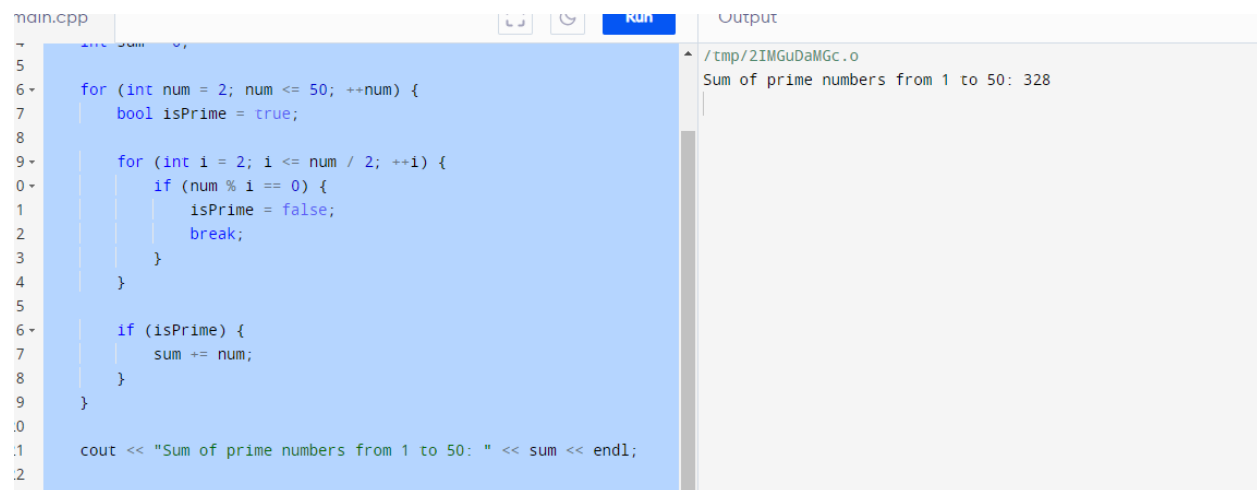
    }
}

if (isPrime) {
    sum += num;
}
}

cout << "Sum of prime numbers from 1 to 50: " << sum << endl;

return 0;
}

```



The screenshot shows a C++ IDE with a file named 'main.cpp'. The code in the editor is as follows:

```

4 int sum = 0;
5
6 for (int num = 2; num <= 50; ++num) {
7     bool isPrime = true;
8
9     for (int i = 2; i <= num / 2; ++i) {
10        if (num % i == 0) {
11            isPrime = false;
12            break;
13        }
14    }
15
16    if (isPrime) {
17        sum += num;
18    }
19 }
20
21 cout << "Sum of prime numbers from 1 to 50: " << sum << endl;
22
23

```

The IDE has a 'Run' button. To the right, the 'Output' window shows the result of the program execution:

```

/tmp/2IMGuDaMGc.o
Sum of prime numbers from 1 to 50: 328

```

## HOME TASK 02

Write a program in C++ to create the following pattern.

```

1
1 2
1 2 3
1 2 3 4

```



1 2 3 4 5

```
#include <iostream>
```

```
using namespace std;;
```

```
int main() {
```

```
    int rows = 5;
```

```
    for (int i = 1; i <= rows; ++i) {
```

```
        for (int k = 1; k <= i; ++k) {
```

```
            cout << k << " ";
```

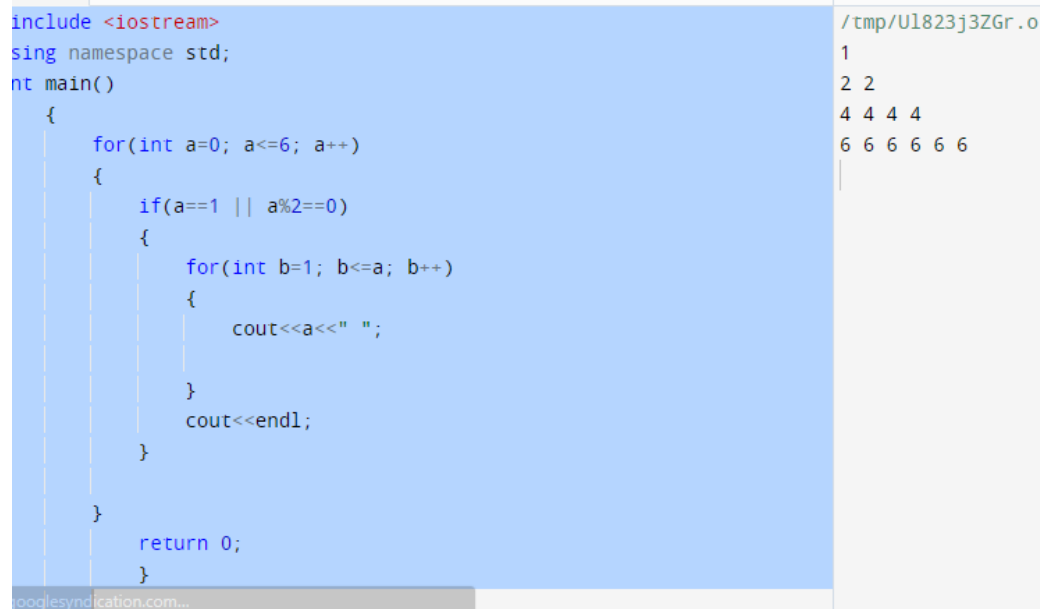
```
        }
```

```
        cout << std::endl;
```

```
    }
```

```
    return 0;
```

```
}
```



The screenshot shows a C++ program in a code editor. The program is designed to print a pattern of numbers. The code is as follows:

```
include <iostream>
sing namespace std;
nt main()
{
    for(int a=0; a<=6; a++)
    {
        if(a==1 || a%2==0)
        {
            for(int b=1; b<=a; b++)
            {
                cout<<a<<" ";
            }
            cout<<endl;
        }
    }
    return 0;
}
```

The output of the program is displayed on the right side of the editor, showing a pattern of numbers:

```
/tmp/Ul823j3ZGr.o
1
2 2
4 4 4 4
6 6 6 6 6 6
```

## HOME TASK 03

Write a C++ program to print:



Edit with WPS Office

1

2 2

4 4 4 4

6 6 6 6 6 6

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    for(int a=0; a<=6; a++)
```

```
    {
```

```
        if(a==1 || a%2==0)
```

```
        {
```

```
            for(int b=1; b<=a; b++)
```

```
            {
```

```
                cout<<a<<" ";
```

```
            }
```

```
            cout<<endl;
```

```
        }
```

```
    }
```

```
    return 0;
```

```
}
```



```
include <iostream>
using namespace std;
int main()
{
    for(int a=0; a<=6; a++)
    {
        if(a==1 || a%2==0)
        {
            for(int b=1; b<=a; b++)
            {
                cout<<a<<" ";
            }
            cout<<endl;
        }
    }
    return 0;
}
```

/tmp/Ul823j3ZGr.o

```
1
2 2
4 4 4 4
6 6 6 6 6 6
```

googlesyndication.com...



Edit with WPS Office