



Prepared By:

Name	CMS ID	Class	Lab Manual
Muhammad Asim Shah	470574	ME-15 "C"	6

Lab Task:

Task 1: Generate the Fibonacci sequence using nested loops.

INPUT:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int num;
```

```
    cout << "Enter the number up to which you want to generate the Fibonacci sequence: ";
```

```
    cin >> num;
```

```
    int a = 0, b = 1;
```

```
    cout << "Fibonacci Sequence up to " << num << ": " << a << ", " << b << ", ";
```

```
    while (a + b <= num) {
```



```
int next = a + b;

cout << next << ", ";

a = b;

b = next;

}

return 0;

}
```

A screenshot of a C++ program running in a console. The program is a Fibonacci sequence generator. The code is shown in a dark-themed editor with line numbers 1 to 21. The console output shows the program asking for a number, then displaying the Fibonacci sequence up to that number. The input is 8, and the output is the sequence 0, 1, 1, 2, 3, 5, 8. The program finishes with exit code 0. A BANDICAM watermark is visible in the bottom right corner.

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int num;
6     cout << "Enter the number up to which you want to generate the Fibonacci sequence: ";
7     cin >> num;
8
9     int a = 0, b = 1;
10    cout << "Fibonacci Sequence up to " << num << ": " << a << ", " << b << ", ";
11
12    while (a + b <= num) {
13        int next = a + b;
14        cout << next << ", ";
15        a = b;
16        b = next;
17    }
18
19    return 0;
20 }
21
```

input

Enter the number up to which you want to generate the Fibonacci sequence: 8
Fibonacci Sequence up to 8: 0, 1, 1, 2, 3, 5, 8,
...Program finished with exit code 0
Press ENTER to exit console.

BANDICAM



Task 2: Create Pascal's triangle with nested loops.

Inputs:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int rows;
```

```
    cout << "Enter the number of rows for Floyd's Triangle: ";
```

```
    cin >> rows;
```

```
    int number = 1;
```

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

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

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

```
            number++;
```

```
        }
```

```
        cout << endl;
```

```
    }
```



```
return 0;
```

```
}
```

Output:

A screenshot of a C++ program and its execution. The top part shows the source code in a dark-themed editor. The code prompts the user for the number of rows and prints Floyd's Triangle for 7 rows. The bottom part shows the program's output in a black console window, displaying the triangle of numbers. A 'BANDICAM' watermark is visible in the bottom right corner of the output window.

```
1 #include <iostream>
2 using namespace std;
3
4 int main() {
5     int rows;
6
7
8     cout << "Enter the number of rows for Floyd's Triangle: ";
9     cin >> rows;
10
11     int number = 1;
12
13
14     for (int i = 1; i <= rows; i++) {
15         for (int j = 1; j <= i; j++) {
16             cout << number << " ";
17             number++;
18         }
19         cout << endl;
20     }
21
22     return 0;
23 }
24
```

input

```
Enter the number of rows for Floyd's Triangle: 7
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21
22 23 24 25 26 27 28
```

BANDICAM