Nested loop in C++

Nested loop means a loop statement inside another loop statement. That is why nested loops are also called as "**loop inside loop**".

Syntax for Nested For loop:

```
for ( initialization; condition; increment ) {
   for ( initialization; condition; increment ) {
      // statement of inside loop
   }
   // statement of outer loop
}
```

Syntax for Nested While loop:

```
while(condition) {
   while(condition) {
      // statement of inside loop
   }
   // statement of outer loop
}
```

Syntax for Nested Do-While loop:

```
do{
   do{
      // statement of inside loop
   }while(condition);
   // statement of outer loop
}while(condition);
```

> Note: There is no rule that a loop must be nested inside its own type. In fact, there can be any type of loop nested inside any type and to any level.

Syntax:

```
do{
   while(condition) {
      for ( initialization; condition; increment ) {
         // statement of inside for loop
      }
      // statement of inside while loop
   }
   // statement of outer do-while loop
}while(condition);
```

Below are some examples to demonstrate the use of Nested Loops:

Example 1: Below program uses a nested for loop to print a 2D matrix of 3×3.

```
// C++ program that uses nested for loop
// to print a 2D matrix

#include <bits/stdc++.h>
using namespace std;

#define ROW 3
#define COL 3

// Driver program
int main()
{

    int i, j;

    // Declare the matrix
    int matrix[ROW][COL] = { { 1, 2, 3 },
                             { 4, 5, 6 },
                             { 7, 8, 9 } };
    cout << "Given matrix is \n";

    // Print the matrix using nested loops
    for (i = 0; i < ROW; i++) {
```

```
        for (j = 0; j < COL; j++)
            cout << matrix[i][j];

        cout << "\n";
    }

    return 0;
}
```

Output

```
Given matrix is
123
456
789
```

Example 2: Below program uses a nested for loop to print all prime factors of a number.

```
// C++ Program to print all prime factors
// of a number using nested loop

#include <bits/stdc++.h>
using namespace std;

// A function to print all prime factors of a given number n
void primeFactors(int n)
{
    // Print the number of 2s that divide n
    while (n % 2 == 0) {
        cout << 2;
        n = n / 2;
    }

    // n must be odd at this point. So we can skip
    // one element (Note i = i +2)
    for (int i = 3; i <= sqrt(n); i = i + 2) {
        // While i divides n, print i and divide n
        while (n % i == 0) {
            cout << i;
```

```
            n = n / i;
        }
    }

    // This condition is to handle the case when n
    // is a prime number greater than 2
    if (n > 2)
        cout << n;
}

/* Driver program to test above function */
int main()
{
    int n = 315;
    primeFactors(n);
    return 0;
}
```

Output

```
3357
```