Nested Loops

Loops within loops are termed as nested loops.

Pattern Problems

The easiest way to understand the working of a nested loop is by solving pattern printing problems.

Print a solid rectangle

```
#include <iostream>

using namespace std;

int main() {
   int rows, columns;
   cout << "Enter the number of rows :\n";
   cin >> rows;
   cout << "Enter the number of columns :\n";
   cin >> columns;

   for (int i = 1; i <= rows; i++) {
       for (int j = 1; j <= columns; j++) {
           cout << "*";
       }
       cout << endl;
   }
   return 0;
}
```

Print a hollow rectangle

```
#include <iostream>

using namespace std;

int main() {
   int rows, columns;
   cout << "Enter the number of rows :\n";
   cin >> rows;
   cout << "Enter the number of columns :\n";
   cin >> columns;

   for (int i = 1; i <= rows; i++) {
       for (int j = 1; j <= columns; j++) {
           if (i == 1 || i == rows || j == 1 || j == columns) {
               cout << "*";
           } else {
               cout << " ";
           }
       }
       cout << endl;
   }
   return 0;
}
```

Print Half Pyramid Pattern using Stars

```
*
**
***
****
*****
```

```
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   for (int i = 1; i <= n; i++) {
       for (int j = 1; j <= i; j++) {
           cout << "*";
       }
       cout << endl;
   }
   return 0;
}
```

Print an Inverted Half pyramid

```
* * * * *
* * * *
* * *
* *
*
```

```
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;
```

```
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= i; j++) {
            cout << "* ";
        }
        cout << endl;
    }
    return 0;
}
```

Print Half Pyramid after 180 degree rotation

```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if (j <= n - i) {
                cout << " ";
            } else {
                cout << "*";
            }
        }
        cout << endl;
    }
    return 0;
}
```

Print Half Pyramid using numbers

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   for (int i = 1; i <= n; ++i) {
       for (int j = 1; j <= i; ++j) {
           cout << j << " ";
       }
       cout << "\n";
   }
   return 0;
}
```

Print Inverted Half Pyramid using numbers

```
1 2 3 4 5
1 2 3 4
1 2 3
1 2
1
```

```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   for (int i = n; i >= 1; --i) {
       for (int j = 1; j <= i; ++j) {
           cout << j << " ";
       }
       cout << endl;
   }
   return 0;
}
```

Print Half Pyramid using numbers - 2

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
```

```cpp
#include <iostream>

using namespace std;

int main() {
```

```
    int n;
    cin >> n;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; ++j) {
            cout << i << " ";
        }
        cout << endl;
    }
    return 0;
}
```

Print Inverted Half Pyramid using numbers - 2

```
1 1 1 1 1
2 2 2 2
3 3 3
4 4
5
```

```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i + 1; ++j) {
            cout << i << " ";
        }
        cout << endl;
    }
    return 0;
}
```

Print Floyd's Triangle

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   int number = 1;
   for (int i = 1; i <= n; i++) {
       for (int j = 1; j <= i; j++) {
           cout << number << " ";
           number++;
       }
       cout << endl;
   }
   return 0;
}
```

Print 0-1 pattern

```
1
0 1
1 0 1
0 1 0 1
1 0 1 0 1
```

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

int main(){

    int i,j,r;

    cin>>r;

    for(i=1;i<=r;i++)
    {
        for(j=1;j<=i;j++)
        {
            if((i+j)%2==0)
                cout<<" 1";
            else
                cout<<" 0";
        }
        cout<<endl;
    }

    return 0;
}
```

Pattern - 5 Palindromic pattern

```
        1
      2 1 2
    3 2 1 2 3
  4 3 2 1 2 3 4
5 4 3 2 1 2 3 4 5
```

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

int main(){
```

```
int i,j,r;

cin>>r;

for(i=1;i<=r;i++)
{
   int k=i;
   for(j=1;j<=(r-i);j++)
   {
      cout<<"  ";
   }
   for( ;j<=r;j++)
   {
      cout<<k<<" ";
      k--;
   }
   k=1;
   for( ;j<(r+i);j++)
   {
      k++;
      cout<<k<<" ";
   }
   for( ;j<=(2*r-1);j++)
   {
      cout<<"  ";
   }


   cout<<endl;
}


return 0;
```

}

Advanced Pattern Problems

Print Diamond using Stars : Given n, print diamond with 2*n rows.

```
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   int space = (2 * n - 1) / 2;
   for (int i = 1; i <= n; i++) {
```

```
        for (int j = 1; j <= space; j++) {
            cout << " ";
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            cout << "*";
        }
        cout << endl;
        space--;
    }

    space = 0;
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= space; j++) {
            cout << " ";
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            cout << "*";
        }
        cout << endl;
        space++;
    }
    return 0;
}
```

Print Hollow Diamond using Stars

```
#include <iostream>
```

```cpp
using namespace std;

int main() {
    int n;
    cin >> n;

    int space = (2 * n - 1) / 2;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= space; j++) {
            cout << " ";
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            if (j == 1 || j == 2 * i - 1) {
                cout << "*";
            } else {
                cout << " ";
            }
        }
        cout << endl;
        space--;
    }

    space = 0;
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= space; j++) {
            cout << " ";
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            if (j == 1 || j == 2 * i - 1) {
                cout << "*";
            } else {
                cout << " ";
            }
        }
        cout << endl;
        space++;
    }
    return 0;
}
```

Print Hollow Diamond Inscribed in a Rectangle

```
*********
****  ****
***    ***
**      **
*        *
*        *
**      **
***    ***
****  ****
*********
```

```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   int space = (2 * n - 1) / 2;
   for (int i = 1; i <= n; i++) {
       for (int j = 1; j <= space; j++) {
           cout << "*";
       }
       for (int j = 1; j <= 2 * i - 1; j++) {
           if (j == 1 || j == 2 * i - 1) {
               cout << "*";
           } else {
               cout << " ";
           }
       }
       for (int j = 1; j <= space; j++) {
           cout << "*";
       }
       cout << endl;
       space--;
   }
```

```
    space = 0;
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= space; j++) {
            cout << "*";
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            if (j == 1 || j == 2 * i - 1) {
                cout << "*";
            } else {
                cout << " ";
            }
        }
        for (int j = 1; j <= space; j++) {
            cout << "*";
        }
        cout << endl;
        space++;
    }
    return 0;
}
```

Print Solid Rhombus

```
     *****
    *****
   *****
  *****
 *****
```

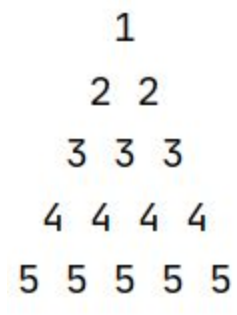
```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }
        for (int j = 1; j <= n; j++) {
```

```
            cout << "*";
        }
        cout << "\n";
    }
    return 0;
}
```

Print Hollow Rhombus

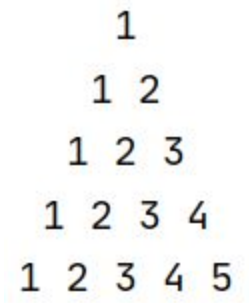
```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;

    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }
        for (int j = 1; j <= n; j++) {
            if (i == 1 || i == n) {
                cout << "*";
            } else {
                if (j == 1 || j == n) {
                    cout << "*";
                } else {
                    cout << " ";
                }
            }
        }
        cout << "\n";
    }
    return 0;
}
```

Pyramid pattern for Numbers




```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   int space = n - 1;
   for (int i = 1; i <= n; i++) {
       for (int j = 1; j <= space; j++) {
           cout << " ";
       }
       space--;
       for (int j = 1; j <= i; j++) {
           cout << i << " ";
       }
       cout << "\n";
   }
   return 0;
}
```

Pyramid Pattern for Numbers - 2




```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   int space = n - 1;
   for (int i = 1; i <= n; i++) {
       for (int j = 1; j <= space; j++) {
           cout << " ";
       }
       space--;
       for (int j = 1; j <= i; j++) {
           cout << j << " ";
       }
       cout << "\n";
   }
   return 0;
}
```

Print Sold Butterfly Pattern

```
*          *
**        **
***      ***
****    ****
**********
**********
****    ****
***      ***
**        **
*          *
```

```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   for (int i = 1; i <= n; i++) {
       int empty_space = 2 * n - 2 * i;
       for (int j = 1; j <= i; j++) {
           cout << "*";
       }
       for (int j = 1; j <= empty_space; j++) {
           cout << " ";
       }
       for (int j = 1; j <= i; j++) {
           cout << "*";
       }
       cout << "\n";
   }

   for (int i = n; i >= 1; i--) {
       int empty_space = 2 * n - 2 * i;
       for (int j = 1; j <= i; j++) {
           cout << "*";
```

```
        }
        for (int j = 1; j <= empty_space; j++) {
            cout << " ";
        }
        for (int j = 1; j <= i; j++) {
            cout << "*";
        }
        cout << "\n";
    }

    return 0;
}
```

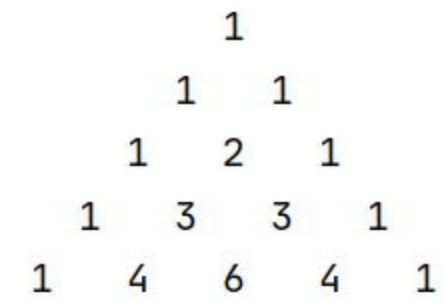
Print Hollow Butterfly Pattern

```
*          *
**        **
* *      * *
*  *    *  *
*   **   *
*   **   *
*  *    *  *
* *      * *
**        **
*          *
```

```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;

    for (int i = 1; i <= n; i++) {
        int empty_space = 2 * n - 2 * i;
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i) {
                cout << "*";
```

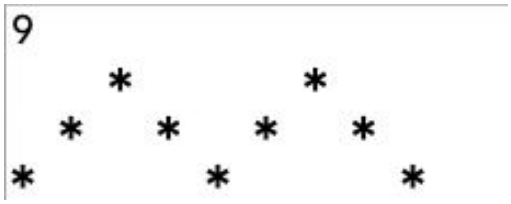
```
            } else {
                cout << " ";
            }
        }
        for (int j = 1; j <= empty_space; j++) {
            cout << " ";
        }
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i) {
                cout << "*";
            } else {
                cout << " ";
            }
        }
        cout << "\n";
    }

    for (int i = n; i >= 1; i--) {
        int empty_space = 2 * n - 2 * i;
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i) {
                cout << "*";
            } else {
                cout << " ";
            }
        }
        for (int j = 1; j <= empty_space; j++) {
            cout << " ";
        }
        for (int j = 1; j <= i; j++) {
            if (j == 1 || j == i) {
                cout << "*";
            } else {
                cout << " ";
            }
        }
        cout << "\n";
    }

    return 0;
}
```

Print Pascal's Triangle




```cpp
#include <iostream>

using namespace std;

int main() {
   int n;
   cin >> n;

   int coef = 1;
   for (int i = 0; i < n; i++) {
       for (int space = 1; space <= n - i; space++)
           cout << "  ";

       for (int j = 0; j <= i; j++) {
           if (j == 0 || i == 0) {
               coef = 1;
           } else {
               coef = coef * (i - j + 1) / j;
           }
           cout << coef << "   ";
       }
       cout << endl;
   }

   return 0;
}
```

Zig-Zag Pattern




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

int main(){
    int i,j,n;

    cin>>n;

    for(i=1;i<=3;i++){

        for(j=1;j<=n;j++){
            if(((i+j)%4==0)||((i==2)&&(j%4==0)))
                cout<<"* ";
            else
                cout<<"  ";
        }
        cout<<endl;
    }

    return 0;
}
```