

Triangle Files

upTriangle

```
//
/* This triangle asks user to enter an add number n. It asks user to to draw
* a triangle with n columns and (n+1)/2 number of rows.
*/

#include <iostream>
using namespace std;

int main() {
    int n;

    cout << "Enter an odd positive number between 1 and 30: ";
    cin >> n;

    if (n<0 || n % 2 == 0)
        return 0;

    for (int r = (n + 1) / 2; r >= 1; r--) {
        for (int c = 1; c <= n; c++) {
            if (c >= r && c + r <= n + 1)
                cout << "*";
            else
                cout << " ";
            /* The condition for for spaces first and stars second
            is the following: Choose the one you like.

            if( c < r || c + r > n + 1)
            cout << " ";
            else
            cout << "*"; */
        }
        cout << endl;
    }
    return 0;
}
```

upTriangle 2

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

int main() {
    int rows;

    cout << "Enter a positive odd integer for the rows: ";
    cin >> rows;

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

Left pointing Triangle (This triangle asks user to enter an odd positive integer between 1 and 30 for the number of rows n. The number of columns are given by $(n+1)/2$ The triangle point to the left

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

int main() {
    int n, mid;

    cout << "Enter a positive odd integer for the rows: ";
    cin >> n;
    mid = (n + 1) / 2;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= mid; j++) {
            if ((i+j < mid + 1) || (i-j > mid - 1))
                cout << " ";
            else
                cout << "*";
        }
        cout << endl;
    }
    return 0;
}
```

Filled diamond

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

int main() {
    int n, mid;

    cout << "Enter a positive odd integer for the rows: ";
    cin >> n;
    mid = (n + 1) / 2;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            if ((i+j < mid + 1) || (i-j > mid - 1) ||
                (j-i > mid-1) || (i + j > n + mid))
                cout << " ";
            else
                cout << "*";
        }
        cout << endl;
    }
    return 0;
}
```