

1- Write a program to print a simple isosceles triangle with 5 rows.

اكتب برنامجًا لطباعة مثلث متساوي الساقين بسيط مكون من 5 صفوف.

Output

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

Solution

```
// www.gammal.tech  
  
#include <iostream>  
using namespace std;  
  
int main() {  
    for (int i = 1; i <= 5; i++) {  
        for (int j = 1; j <= i; j++) {  
            cout << " * ";  
        }  
        cout << endl;  
    }  
    return 0;  
}
```

2- Write a program to print an inverted isosceles triangle with 4 rows.

اكتب برنامجًا لطباعة مثلث متساوي الساقين مقلوب مكون من أربعة صفوف.

Output

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

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    for (int i = 4; i >= 1; i--) {
        for (int j = 1; j <= i; j++) {
            cout << " * ";
        }
        cout << endl;
    }
    return 0;
}
```

3- Write a program to print a hollow isosceles triangle with 6 rows.

اكتب برنامجا لطباعة مثلث مجوف متساوي الساقين مكون من 6 صفوف.

Output

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

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    int n = 6;
    for (int i = 1; i <= n; i++) {
        for (int j = i; j < n; j++) {
            cout << " ";
        }
        for (int j = 1; j <= 2 * i - 1; j++) {
            if (j == 1 || j == 2 * i - 1 || i == n)
                cout << "*";
            else
                cout << " ";
        }
        cout << endl;
    }
    return 0;
}
```

4- Write a program to print a mirrored isosceles triangle with 3 rows.

اكتب برنامجا لطباعة مثلث متساوي الساقين معكوس مكون من ثلاثة صفوف.

Output

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

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    int n = 3;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < i; j++) {
            cout << " ";
        }
        for (int j = 0; j < n - i; j++) {
            cout << "* ";
        }
        cout << endl;
    }
    return 0;
}
```

5- Write a program to print an isosceles triangle of numbers with 4 rows.

اكتب برنامجًا لطباعة مثلث متساوي الساقين مكون من أربعة صفوف.

Output

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

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    for (int i = 1; i <= 4; i++) {
        for (int j = 1; j <= i; j++) {
            cout << j << " ";
        }
        cout << endl;
    }
    return 0;
}
```

6- Write a program to print a rotated isosceles triangle with 5 rows.

اكتب برنامجًا لطباعة مثلث متساوي الساقين مدور مكون من 5 صفوف.

Output

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

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    int n = 5;
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }
        for (int j = 1; j <= i; j++) {
            cout << "* ";
        }
        cout << endl;
    }
    return 0;
}
```

7- Write a program to print a pattern of letters forming an isosceles triangle with 4 rows.

اكتب برنامجًا لطباعة نمط من الحروف يشكل مثلثًا متساوي الساقين مكون من 4 صفوف.

Output

```
A
A B
A B C
A B C D
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    int n = 4;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }
        for (int j = 1; j <= i; j++) {
            cout << char('A' + j - 1) << " ";
        }
        cout << endl;
    }
    return 0;
}
```

8- Write a program to print an inverted isosceles triangle of letters with 5 rows.

اكتب برنامجاً لطباعة مثلث متساوي الساقين مقلوب مكون من 5 صفوف.

Output

```
E E E E E
D D D D
C C C
B B
A
```

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    int n = 5;
    for (int i = n; i >= 1; i--) {
        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }
        for (int j = 1; j <= i; j++) {
            cout << char('A' + i - 1) << " ";
        }
        cout << endl;
    }
    return 0;
}
```

9- Create a program that takes an input for the number of rows and generates Pascal's Triangle up to the specified number of rows. Output the triangle with appropriate spacing.

قم بإنشاء برنامج يأخذ مدخلات لعدد الصفوف ويقوم بإنشاء مثلث باسكال حتى العدد المحدد من الصفوف. إخراج المثلث مع التباعد المناسب.

Input

```
Enter number of rows: 5
```

Output

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

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {

    // Declare variables
    int rows, coef = 1;

    // Input the number of rows from the user
    cout << "Enter number of rows: ";
    cin >> rows;

    // Loop to generate Pascal's Triangle
    for(int i = 0; i < rows; i++) {
        // Print leading spaces
        for(int space = 1; space <= rows - i; space++)
            cout << " ";

        // Calculate and print coefficients for the current row
        for(int j = 0; j <= i; j++) {
            if (j == 0 || i == 0)
                coef = 1;
            else
                coef = coef * (i - j + 1) / j;

            cout << coef << " ";
        }
        // Move to the next line after completing a row
        cout << endl;
    }

    return 0;
}
```

10- Develop a straightforward program that generates a diamond pattern based on the user-provided number of rows. Keep the program simple and easy to comprehend.

قم بتطوير برنامج مباشر يقوم بإنشاء نمط معين بناءً على عدد الصفوف التي يقدمها المستخدم. اجعل البرنامج بسيطاً وسهل الفهم.

Input

```
Enter the number of rows for the diamond: 5
```


Output

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

Solution

```
// www.gammal.tech

#include <iostream>
using namespace std;

int main() {
    int rows;

    // Input the number of rows from the user
    cout << "Enter the number of rows for the diamond: ";
    cin >> rows;

    // Generate diamond pattern
    for (int i = 1; i <= rows; i++) {
        // Print leading spaces
        for (int space = 1; space <= rows - i; space++)
            cout << " ";

        // Print asterisks for the first half of the diamond
        for (int j = 1; j <= 2 * i - 1; j++)
            cout << "* ";

        cout << endl;
    }

    for (int i = rows - 1; i >= 1; i--) {
        // Print leading spaces
        for (int space = 1; space <= rows - i; space++)
            cout << " ";

        // Print asterisks for the second half of the diamond
        for (int j = 1; j <= 2 * i - 1; j++)
            cout << "* ";

        cout << endl;
    }

    return 0;
}
```