

TRIANGULO DE PASCAL

RESULTADO:

```
joyita@DESKTOP-UH3QJPI:~/Ej_2do_Corte$ ./a.out
Ingrese el número de filas del Triángulo de Pascal: 4
  1
 1 1
1 2 1
1 3 3 1

joyita@DESKTOP-UH3QJPI:~/Ej_2do_Corte$ ./a.out
Ingrese el número de filas del Triángulo de Pascal: 10
      1
     1 1
    1 2 1
   1 3 3 1
  1 4 6 4 1
 1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1
1 8 28 56 70 56 28 8 1
1 9 36 84 126 126 84 36 9 1

joyita@DESKTOP-UH3QJPI:~/Ej_2do_Corte$
```

METODOLOGIA:

Para crear el código del Triángulo de Pascal, primero quise entender cómo se forma este triángulo de números. Vi que, en cada fila, los números de los extremos siempre son 1 y que los del medio se obtienen sumando los dos números que están justo encima de ellos en la fila anterior.

Con esa idea, empecé a construir el programa. Definí una clase llamada TrianguloPascal que recibe la cantidad de filas que el usuario desea. Luego, dentro de un método, fui llenando las filas una por una: primero asignando los bordes con 1 y después calculando los valores del centro sumando los números de la fila anterior.

Y ya por último, hice una función para mostrar el triángulo de forma ordenada en la pantalla, dejando espacios para que tuviera una forma parecida a la del triángulo real. Probé el programa con varios números de filas hasta que el resultado se veía bien alineado y correcto.

DIFICULTADES:

Al inicio me costó un poco entender cómo se generaba el triángulo fila por fila y cómo acceder a los valores de la fila anterior para hacer las sumas. También me confundí al principio con la forma de imprimir el triángulo, ya que los números quedaban desordenados o muy juntos por lo que tuve que hacer varias pruebas.

. Otra dificultad fue recordar que las primeras y últimas posiciones de cada fila siempre debían ser 1 si no se tenía en cuenta pues los resultados salían incorrectos.