

Final II Término 2003-2004. Febrero 10, 2004

Tema 1 (20 puntos) Escriba un programa en C/C++ que permita almacenar en una matriz las primeras n filas, $n < 20$, del triángulo de pascal.

Muestre por pantalla la matriz generada.

1	0	0	0	0	0
1	1	0	0	0	0
1	2	1	0	0	0
1	3	3	1	0	0
1	4	6	4	1	0
1	5	10	10	5	1
...

En la matriz se muestra un ejemplo. A partir de la tercera fila, los elementos dentro del triángulo, se obtienen sumando los dos elementos anteriores de la fila anterior. Por ejemplo, $6=3+3$.

Propuesta de Solución:

Para la solución se usará una matriz cuadrada de tamaño $n \times n$. Al recorrer la matriz, siempre que se encuentre en la primera columna o en la diagonal se escribe 1, sino se suma los valores de dos casillas superiores. Se recorre toda la matriz y se presenta el resultado.

Tarea: Realizar el ejercicio usando lazos "para"

Descripción	Python
Inicio	<pre># ICM00794-Fundamentos de Computación - FCNM-ESPOL # 3ra Eval II Termino 2006 # Tema 2. Triangulo pascal # propuesta: edelros@espol.edu.ec</pre>
Requiere usar librería numpy Ingresa el tamaño del triángulo	<pre>import numpy n=int(input('tamaño del triangulo: '))</pre>
Inicializa matriz pascal Se usará la fila 0 Mientras f esté dentro de la matriz Se usa columna desde 0 Mientras c este debajo de la diagonal Si es la primera columna o es la diagonal Se escribe 1 Sino se suma de las casillas de arriba cambia columna cambia fila	<pre>#Procedimiento pascal=numpy.zeros(shape=(n,n),dtype=int) f=0 while (f<n): c=0 while (c<=f): if (c==0 or c==f): pascal[f,c]=1 else: pascal[f,c]=pascal[f-1,c]+pascal[f-1,c-1] c=c+1 f=f+1</pre>
muestra la matriz pascal	<pre>#Salida print(pascal)</pre>

Ejecución del algoritmo: triangulopascal.py

<pre>>>> tamaño del triangulo: 5 [[1 0 0 0] [1 1 0 0] [1 2 1 0] [1 3 3 1] [1 4 6 4]]</pre>	<pre>tamaño del triangulo: 7 [[1 0 0 0 0 0 0] [1 1 0 0 0 0 0] [1 2 1 0 0 0 0] [1 3 3 1 0 0 0] [1 4 6 4 1 0 0] [1 5 10 10 5 1 0] [1 6 15 20 15 6 1]]</pre>
---------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------