

Exercício: Triângulo de Pascal

1 Descrição

O Triângulo de Pascal é uma matriz triangular de coeficientes binomiais. Ficou conhecido por este nome pelo resultado dos trabalhos do matemático francês Blaise Pascal, que estudou as propriedades do triângulo. Um dos usos mais conhecidos do Triângulo de Pascal é a determinação dos coeficientes em uma expansão binomial. Por exemplo, os coeficientes da expansão de $(x + y)^N$, são os números da linha $N + 1$ do Triângulo de Pascal. Veja:

$$(x + y)^2 = 1 \times x^2 + 2 \times x \times y + 1 \times y^2$$

Os coeficientes 1,2,1 fazem parte da linha 3 do triângulo.

Agora análise as seis primeiras linhas do Triângulo de Pascal:

$$\begin{bmatrix} 1 \\ 1 & 1 \\ 1 & 2 & 1 \\ 1 & 3 & 3 & 1 \\ 1 & 4 & 6 & 4 & 1 \\ 1 & 5 & 10 & 10 & 5 & 1 \end{bmatrix}$$

Como você pode notar, a primeira linha do Triângulo de Pascal é formada apenas pelo número 1. As linhas a seguir formam uma sequência. Existem diversas maneiras de se calcular o Triângulo de Pascal, pesquise e implemente um programa em C que leia na entrada padrão de texto um número real N . Em seguida, imprima na tela todas as linhas do Triângulo de Pascal até a linha N . Você não deve imprimir espaços entre os números de uma mesma linha.

2 Instruções Complementares

- Apesar do resultado o Triângulo de Pascal formar uma matriz, não é necessário utilizar uma estrutura de dados de matriz para resolver este exercício, pense (e pesquise, se necessário) para solucionar este exercício apenas com variáveis simples (inteiros)
- Pesquise sobre a biblioteca `math.h` e qual função pode ser utilizada para realizar esta operação
- Mais informações sobre o Triângulo de Pascal: http://pt.wikipedia.org/wiki/Tri%C3%A2ngulo_de_Pascal
- Submeta o arquivo `.c` com seu código no <http://run.codes>

3 Exemplos de Entrada e Saída

A seguir são apresentados exemplos de entrada e saída para que você teste seu código enquanto desenvolve o exercício. Este são apenas exemplos ilustrativos, somente uma pequena parte das operações está representada. Enquanto estiver desenvolvendo, elabore novos testes para validar seu código.

Entrada

2

Saída

1

11

Entrada

6

Saída

1

11

121

1331

14641

161051