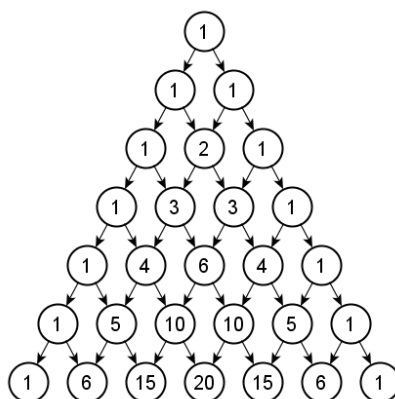


(O texto a seguir se refere às questões 1, 2 e 3)

O triângulo de Pascal é uma estrutura numérica com algumas propriedades peculiares. Em um triângulo de Pascal, a primeira linha tem 1 elemento, a segunda tem 2 elementos, a terceira tem 3 elementos, e assim por diante. O primeiro e o último elemento de cada linha têm valor 1. Os demais elementos podem ser obtidos somando-se os valores que aparecem imediatamente acima de cada posição na linha anterior. Por exemplo, a figura abaixo mostra o triângulo de Pascal de grau 7:



Um triângulo de Pascal pode ser representado como uma matriz, alinhando os elementos de cada linha à esquerda, e preenchendo com 0 as posições não usadas. Por exemplo, o triângulo acima pode ser representado pela seguinte matriz 7x7:

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

1. Escreva uma função que aloca uma matriz com grau x grau posições e a preenche de forma a gerar um triângulo de Pascal como descrito acima, retornando o endereço da matriz alocada ao final. O protótipo da função deve ser o seguinte:

```
int** geraTrianguloDePascal (int grau);
```

2. A função da questão anterior deve alocar uma matriz, mas não deve desalocá-la.

a) Por que a matriz não deve ser desalocada na função?

b) Escreva uma função que desaloca a matriz. A função deve ter o seguinte protótipo:

```
void destroiTrianguloDePascal (int** t, int grau);
```

3. Suponha que o triângulo de Pascal de grau 8 (i.e com uma linha a mais que o exemplo do início!) foi gerado, e em seguida a função abaixo foi chamada. O que será mostrado pela função? Não é preciso explicar a saída, apenas mostre o que será impresso.

```
0 void misterio (int** t, int grau)
1 {
2     int a, b, c, d;
3
4     for (a = 0; a < grau; a++)
5     {
6         c = 0;
7         d = 0;
8         for (b = 0; b < grau; b++)
9             c += t [a][b];
10        for (b = 0; b <= a; b++)
11            d += t [a-b][b];
12        printf ("%d\t%d\n", c, d); /* \t: imprime uma tabulação. */
13    }
14 }
```