

1.

```
int** geraTrianguloDePascal (int grau)
{
    int** m;
    int i, j;

    // Aloca.
    m = (int**) malloc (sizeof (int*) * grau);
    for (i = 0; i < grau; i++)
        m [i] = (int*) malloc (sizeof (int) * grau);

    // Inicializa a primeira linha com 1000... e a primeira coluna com 1s.
    for (i = 1; i < grau; i++)
        m [0][i] = 0;
    for (i = 0; i < grau; i++)
        m [i][0] = 1;

    // Preenche o resto. Basta olhar para as posições j-1 e j da linha acima!
    for (i = 1; i < grau; i++)
        for (j = 1; j < grau; j++)
            m [i][j] = m [i-1][j-1] + m [i-1][j];

    return (m);
}
```

Alternativa para o bloco central:

```
for (i = 0; i < grau; i++)
{
    m [i][0] = 1;
    m [i][i] = 1;

    for (j = 1; j < i; j++)
        m [i][j] = m [i-1][j-1] + m [i-1][j];

    for (j = i+1; j < grau; j++)
        m [i][j] = 0;
}
```

2.

a) Se a matriz fosse desalocada na função, ela deixaria de existir fora da função, e não poderia ser usada para nada!

b)

```
void destroiTrianguloDePascal (int** t, int grau)
{
    int i;
    for (i = 0; i < grau; i++)
        free (t [i]);
    free (t);
}
```

3. É muito estranho, mas a primeira coluna mostra as potências de 2, e a segunda, a sequência de Fibonacci!

1	1
2	1
4	2
8	3
16	5
32	8
64	13
128	21