

Quadtree II

Por Mark Dettinger  Alemanha

Timelimit: 1

Quando percebeu que o mapa do tesouro codificado em quadtree era uma falsificação, Florida Jones fez um plano maligno para pregar uma peça no próximo caçador de tesouros que viesse atrás do tesouro. Mas para isso, ele precisa da sua ajuda novamente.

Você pode escrever um programa que use uma figura no formato XBM e *codifica-la* no esquema quadtree?

Entrada

- A primeira linha será “#define quadtree_width **N**”, onde **N** é o tamanho da figura em pixels. (A figura é quadrática, contendo **N*N** pixels).
 - A segunda linha será “define quadtree_height **N**”, concordando com a primeira.
 - A terceira linha será “#define quadtree_bits[] = {”.
 - Então seguirão **N** linhas, cada uma codificando uma linha de pixels na figura. Haverão **N/8** números hexadecimais por linha.
Cada número hexadecimal é composto por 8 bits que codificam 8 pixels da esquerda para a direita (onde o bit da extrema esquerda tem valor 1 e o bit da extrema direita tem valor 128). Os números hexadecimais são impressos no formato 0xdd, onde *d* é um dos caracteres do conjunto { 0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f }.
- Exemplo: Os 8 pixels WBBBWWB são escritos como 0x9e. (2 + 4 + 8 + 16 + 128 = 158 = 0x9e)
- A última linha será “};”.

Nota: Os comentários no exemplo de entrada (delimitados por /* e */) não são parte da entrada. Eles devem ajudar a explicar o formato **XBM**.

Saída

Primeiro, imprima o inteiro **N** ($8 \leq N \leq 512$) em uma única linha. Então, imprima uma string de letras **B**, **W** e **Q** que corretamente codificam a figura com o esquema quadtree. Finalmente, acabe a string com um caractere de nova linha.

Exemplo de Entrada	Exemplo de Saída
<pre>#define quadtree_width 16 #define quadtree_height 16 static char quadtree_bits[] = { 0xf0,0xf0, WWWBBBB WWWBBBB */ 0xf0,0xf0, WWWBBBB WWWBBBB */ 0xf0,0xf0, WWWBBBB WWWBBBB */ 0xf0,0xf0, WWWBBBB WWWBBBB */ 0x0f,0x0f, BBBBWWWW BBBBWWWW */ 0x0f,0x0f, BBBBWWWW BBBBWWWW */ 0x0f,0x0f,</pre>	<pre>16 QQWBBWQWBBWQWBBWQWBBW</pre>

BBBBWWWW	BBBBWWWW	*/	
0x0f,0x0f,		/*	
BBBBWWWW	BBBBWWWW	*/	
0xf0,0xf0,		/*	
WWWWBBBB	WWWWBBBB	*/	
0xf0,0xf0,		/*	
WWWWBBBB	WWWWBBBB	*/	
0xf0,0xf0,		/*	
WWWWBBBB	WWWWBBBB	*/	
0xf0,0xf0,		/*	
WWWWBBBB	WWWWBBBB	*/	
0x0f,0x0f,		/*	
BBBBWWWW	BBBBWWWW	*/	
0x0f,0x0f,		/*	
BBBBWWWW	BBBBWWWW	*/	
0x0f,0x0f,		/*	
BBBBWWWW	BBBBWWWW	*/	
0x0f,0x0f,		/*	
BBBBWWWW	BBBBWWWW	*/	
};			