1999/2000 ACM International Collegiate Programming Contest
University of Ulm Local Contest

Problem C: Quadtree

Source file: quadtree.(c|C|pas|java)

While searching for treasures in an ancient Aztec ruin, Florida Jones (the brother of famous Indiana Jones) stumbles across a papyrus roll lettered with a long string of symbols. There are three different symbols occuring in the string which we will call *B*, *W* and *Q* here.
Being somewhat experienced in cryptography, Florida Jones recognizes the code immediately as the famous Quadtree Encoding Scheme that has been invented 3000 years ago.

With the Quadtree system, secret pictures (like treasure maps) were encoded in the following way: If the whole picture was black, it was encoded by the letter *B*. If it was completely white, it was encoded by *W*. If both colors were used (what was usually the case), it was encoded by *Qxxxx* where each *x* was a string that recursively encoded one quarter of the picture (in the order top left, top right, bottom left, bottom right). As the Aztecs always used quadratic pictures with *n*n* pixels where *n* was a power of two, this method always worked perfectly.
A 2*2 chess board, for instance, would be encoded as QWBBW, a 4*4 chess board as QQWBBWQWBBWQWBBWQWBBW.

Your job is to decode the quadtree string and output the picture in the XBM format (see output specification).

Input Specification

The input file contains an integer *n* (8 <= *n* <= 512) on the first line, giving the size of the picture in pixels per row/column. *n* will always be a power of two.
On the second line, a string consisting of the letters B, W and Q follows. The string encodes a picture with *n*n* pixels with the quadtree scheme.

Output Specification

- On the first line, print "#define quadtree_width *n*" where *n* is the picture size given in the input file.
- On the second line, print "#define quadtree_height *n*" accordingly.
- On the third line, print "static char quadtree_bits() = {".
- Then, print *n* lines (each one encoding one pixel row of the picture) with *n/8* hexadecimal numbers per line.
  Each hexadecimal number is composed of 8 bits that encode 8 pixels

from left to right (where the leftmost bit has the value 1 and the rightmost bit has the value 128). The hexadecimal numbers should be printed in the form 0x*dd* where *d* is one character of the set { 0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f }.
Example: The 8 pixels WBBBBWWB would be written as 0x9e. (2+4+8+16+128 = 158 = 0x9e)
Print a comma after each hexadecimal number.

- On the last line, print "};".

Sample Input

```
16
QQWBBWQWBBWQWBBWQWBBW
```

Sample Output

Note: The comments (enclosed by /* and */) are not part of the output. They should help to explain the XBM format.

```
#define quadtree_width 16
#define quadtree_height 16
static char quadtree_bits() = {
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 */
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 */
};
```

Hint

The unix tool *xv* can display graphics in the XBM format.
So, after you have finished your program, run it with "`quadtree | xv -`" and have a look at the picture. This may help in debugging.

Notice

The real XBM format uses square brackets "[" and "]" instead of "(" and ")". It has been changed because of automatic judge limitations.