



4174 - Steganography

North America - Mid Central - 2008/2009

Problem H: Steganography

Source file: `steg.{c,cpp,java}`

Input file: `steg.in`

In cryptography, the goal is to encrypt a message so that, even if the message is intercepted, only the intended recipient can decrypt it. In steganography, which literally means "hidden writing", the goal is to hide the fact that a message has even been sent. It has been in use since 440 BC. Historical methods of steganography include invisible inks and tattooing messages on messengers where they can't easily be seen. A modern method is to encode a message using the least-significant bits of the RGB color values of pixels in a digital image.

For this problem you will uncover messages hidden in plain text. The spaces within the text encode bits; an odd number of consecutive spaces encodes a 0 and an even number of consecutive spaces encodes a 1. The four texts in the example input below (terminated by asterisks) encode the following bit strings: 11111, 000010001101101, 01, and 00010001010001001111. Each group of five consecutive bits represents a binary number in the range 0–31, which is converted to a character according to the table below. If the last group contains fewer than five bits, it is padded on the right with 0's.

" " (space)	0
"A" – "Z"	1–26
"'" (apostrophe)	27
", " (comma)	28
"-" (hyphen)	29
". " (period)	30
"?" (question mark)	31

The first message is $11111_2 = 31_{10} = \text{"?"}$. The second message is $(00001, 00011, 01101)_2 = (1, 3, 13)_{10} = \text{"ACM"}$. The third message is $01\underline{000}_2 = 8_{10} = \text{"H"}$, where the underlined 0's are padding bits. The fourth message is $(00010, 00101, 00010, 01111, \underline{10000})_2 = (2, 5, 2, 15, 16)_{10} = \text{"BEBOP"}$.

Input: The input consists of one or more texts. Each text contains one or more lines, each of length at most 80 characters, followed by a line containing only "*" (an asterisk) that signals the end of the text. A line containing only "#" signals the end of the input. In addition to spaces, text lines may contain any ASCII letters, digits, or punctuation, except for "*" and "#", which are used only as sentinels.

Output: For each input text, output the hidden message on a line by itself. Hidden messages will be 1–64 characters long.

Note: Input text lines and output message lines conform to all of the whitespace rules listed in item 7 of *Notes to Teams* except that there may be consecutive spaces *within* a line. There will be no spaces at the beginning or end of a line.

Example input:	Example output:
<p>Programmer, I would like to see a question mark. *</p> <p>Behold, there is more to me than you might think when you read me the first time. *</p> <p>Symbol for hydrogen? *</p> <p>A B C D E F G H I J K L M N O P Q R S T U V *</p> <p>#</p>	<p>? ACM H BEBOP</p>

Last modified on October 21, 2008 at 9:17 PM.

Mid Central 2008-2009