

Huffman Coding

Input: standard input

Output: standard output

Huffman coding is a data encoding technique where each and every data item is represented as a bit sequence (code word) of varying length. No bit-sequence of a data item is a prefix of bit-sequence of another data item. Individual symbols which makes a message are represented (encoded) with bit sequences (code word) that have been predefined. So that the most frequent items have the shortest bit sequences. This way, storage requirement is reduced compared to fixed-length bit sequences, if the frequency distribution is appropriate for the input data. The following is a code word table (bit sequence) for all uppercase English letters.

Symbol	Code word	Symbol	Code word	Symbol	Code word
A	00	J	11001000	S	1101110
B	010	K	11001001	T	11011111
C	011	L	110011	U	11011010
D	1110	M	11001010	V	11011000
E	10	N	110001	W	110000000
F	11000010	O	11001011	X	110000001
G	11000011	P	11011011	Y	110110010
H	11010	Q	11000001	Z	110110011
I	1111	R	11011110		

If we consider the above code word table then the sequence of bits 0001111100001000 is the encoded bit sequence for the data string ACDABA. Now, you are to write a program to find the encoded bit sequence for a given string.

[Symbol Code word: A 00 B 010 C 011 D 1110 E 10 F 11000010 G 11000011 H 11010 I 1111 J 11001000 K 11001001 L 110011 M 11001010 N 110001 O 11001011 P 11011011 Q 11000001 R 11011110 S 1101110 T 11011111 U 11011010 V 11011000 W 110000000 X 110000001 Y 110110010 Z 110110011]

Input Format

There will be several lines in the input terminated with a line containing a single #. This last line should not be processed. Each of the line will have a string consists of uppercase letters only.

Constraints

- Length of each string will be less than 100.

Output Format

For each line in the input, print the encoded bit sequence for the corresponding line's input string.

Sample Input 0

```
UD
PC
#
```

Sample Output 0

110110101110
11011011011