Statement is not available.

Statement is not available.

Statement is not available.

D. Draw Brackets!

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

A sequence of square brackets is regular if by inserting symbols "+" and "1" into it, you can get a regular mathematical expression from it. For example, sequences "[[]][]", "[]" and "[[][[]]]" — are regular, at the same time "][", "[[]" and "[[]][" — are irregular.

Draw the given sequence using a minimalistic pseudographics in the strip of the lowest possible height — use symbols '+', '-' and '|'. For example, the sequence "[[][]][]" should be represented as:

```
+-          -++-  -+
|+- -++-  -+||     |
||    ||   |||     |
|+- -++-  -+||     |
+-          -++-  -+
```

Each bracket should be represented with the hepl of one or more symbols '|' (the vertical part) and symbols '+' and '-' as on the example which is given above.

Brackets should be drawn without spaces one by one, only dividing pairs of consecutive pairwise brackets with a single-space bar (so that the two brackets do not visually merge into one symbol). The image should have the minimum possible height.

The enclosed bracket is always smaller than the surrounding bracket, but each bracket separately strives to maximize the height of the image. So the pair of final brackets in the example above occupies the entire height of the image.

Study carefully the examples below, they adequately explain the condition of the problem. Pay attention that in this problem the answer (the image) is unique.

Input

The first line contains an even integer n $(2 \leq n \leq 100)$ — the length of the sequence of brackets.

The second line contains the sequence of brackets — these are n symbols "[" and "]". It is guaranteed that the given sequence of brackets is regular.

Output

Print the drawn bracket sequence in the format which is given in the condition. Don't print extra (unnecessary) spaces.

Examples

input
8
[[][]][]

output
| ```
+-          -++-  -+
|+- -++-  -+||     |
||    ||   |||     |
|+- -++-  -+||     |
+-          -++-  -+
``` |

input
6
[[[]]]

output
| ```
+-        -+
|+-      -+|
||+-  -+||
|||    |||
||+-  -+||
|+-      -+|
+-        -+
``` |

input
6
[[][]]

output
| ```
+-          -+
|+- -++-  -+|
||    ||   ||
|+- -++-  -+|
+-          -+
``` |

input

```
2
[]
```

output

```
+-  -+
|    |
+-  -+
```

input

```
4
[][]
```

output

```
+-  -++-  -+
|    ||    |
+-  -++-  -+
```