

---

## Reverse Code

During your recent trip to the beach, you found a glass bottle washed ashore with a slip of paper inside. At first the message inscribed on it seemed unreadable, but you quickly realized it was written in a rather primitive secret code – sections of text between square brackets (“[]”) are simply meant to be read backwards, for example “alg[tiro]hm” actually means “algorithm”.

You can decode enough to learn that the message contains information about hidden pirate treasure, but soon the process becomes too tiresome to perform manually, as the square brackets recursively nest within each other multiple times. You decide that this is clearly a task for a computer.

### Task

Write a program that will decode the secret message by reversing text between square brackets. The message may contain nested brackets (that is, brackets within brackets, such as “One[owT[Three[ruoF]]]”). In this case, innermost brackets take precedence, similar to parentheses in mathematical expressions, e.g. you could decode the aforementioned example like this:

1. One[owT[Three[ruoF]]]
2. One[owT[ThreeFour]]
3. One[owTruoFeerhT]
4. OneThreeFourTwo

In order to make your own task slightly easier and less tricky, you have already replaced all whitespaces in the original text with underscores (“\_”) while copying it from the paper version.

### Input description

The first and only line of the standard input consists of a non-empty string of up to  $2 \cdot 10^6$  characters which may be letters, digits, basic punctuation (“, . ? ! ’ - ; : ”), underscores (“\_”) and square brackets (“[]”). You can safely assume that all square brackets are paired correctly, i.e. every opening bracket has exactly one closing bracket matching it and vice versa.

### Output description

The standard output should contain one line – the decoded secret message without any square brackets.



## Example

For sample input:

```
A[W_[y, []]oh]o[d!r] [!]
```

the correct output is:

```
Ahoy,_World!
```

## Explanation

This example contains empty brackets. Of course, an empty string, when reversed, remains empty, so we can simply ignore them. Then, as previously, we can decode this example in stages, first reversing the innermost brackets to obtain “A[W\_,yoh]o[d!r] [!]”. Afterwards, there are no longer any nested brackets, so the remainder of the task is trivial.



## Scoring

If your algorithm solves only some of the test cases, you will be awarded partial points. The table below describes available test groups with additional constraints.

Additional constraints:	Points for the test group:
Input length $\leq 100$	20
Input length $\leq 10\,000$	30
No additional constraints.	50



Biurow Projektu Partnera Wiodącego: Politechnika Łódzka | Wydział Elektrotechniki, Elektroniki, Informatyki i Automatyki |  
ul. Stefanowskiego 22, pokój 14, 90-924 Łódź | tel. (42) 631-28-89, | e-mail: [biuro@cmi.edu.pl](mailto:biuro@cmi.edu.pl) | [www.cmi.edu.pl](http://www.cmi.edu.pl)

Partner Wiodący Projektu



Partnerzy Projektu



Politechnika  
Warszawska



I<sup>math</sup>

Cyfrowy  
3ialog

