Brackets

https://vjudge.net/problem/poj-2955

We give the following inductive definition of a “regular brackets” sequence:

- the empty sequence is a regular brackets sequence,
- if *s* is a regular brackets sequence, then (*s*) and [*s*] are regular brackets sequences, and
- if *a* and *b* are regular brackets sequences, then *ab* is a regular brackets sequence.
- no other sequence is a regular brackets sequence

For instance, all of the following character sequences are regular brackets sequences:

```
(), [], (()), ()[], ()[()]
```

while the following character sequences are not:

```
(, ], )(, ([)], ([(]
```

Given a brackets sequence of characters $a_1 a_2 \ldots a_n$, your goal is to find the length of the longest regular brackets sequence that is a subsequence of *s*. That is, you wish to find the largest m such that for indices $i_1, i_2, \ldots, i_m$ where $1 \le i_1 < i_2 < \ldots < i_m \le n$, $a_{i1} a_{i2} \ldots a_{im}$ is a regular brackets sequence.

Given the initial sequence `([([])]`, the longest regular brackets subsequence is `[([])]`.

Input

The input test file will contain multiple test cases. Each input test case consists of a single line containing only the characters `(`, `)`, `[`, and `]`; each input test will have length between 1 and 100, inclusive. The end-of-file is marked by a line containing the word “end” and should not be processed.

Output

For each input case, the program should print the length of the longest possible regular brackets subsequence on a single line.

Sample

Input	Output
((())) ()()() ([]))[)(([][][) end	6 6 4 0 6