DWITE '07 R2 #4 - All Is Balanced

Time Limit: 2.0s **Memory Limit:** 64M

DWITE Online Computer Programming Contest, November 2007, Problem 4

Parenthesis are important characters in programming – they define the order of operations and organize information. That is, as long as they are properly balanced. Parenthesis are *balanced* when opening and closing brackets match with each other, and are themselves nested within balanced parenthesis.

This is perhaps best illustrated in examples.

- `()` - balanced: opening is matched with a closing
- `(()` - not balanced: one of the opening brackets has no match
- `([)]` - not balanced: the contents inside of `( )` are not balanced

The input will contain 5 lines, each no more than 255 characters long. Valid characters are any of the three parenthesis types: `()`, `[]`, `{}`, and any alphanumeric characters: `a`-`z` `0`-`9`.

The output will contain 5 lines, each stating either `balanced` or `not balanced`, for the supplied expressions.

All of the characters but the parenthesis could effectively be ignored. Keep in mind the different types of parenthesis – an opening bracket needs to be matched with a closing bracket of the same type.

Sample Input

```
abc
([{a}b]c)
)(
([)]
{abc]
```

Sample Output

```
balanced
balanced
not balanced
not balanced
not balanced
```

Problem Resource: DWITE