

Problem B: Clickomania

Source: `clickomania`. {c, cpp, java}

Input: `console` {`stdin`, `cin`, `System.in`}

Output: `console` {`stdout`, `cout`, `System.out`}

Clickomania is a puzzle in which one starts with a rectangular grid of cells of different colours. In each step, a player selects ("clicks") a cell. All connected cells of the same colour as the selected cell (including itself) are removed if the selected cell is connected to at least one other cell of the same colour. The resulting "hole" is filled in by adjacent cells based on some rule, and the object of the game is to remove all cells in the grid.

In this problem, we are interested in the one-dimensional version of the problem. The starting point of the puzzle is a string of colours (each represented by an uppercase letter). At any point, one may select (click) a letter provided that the same letter occurs before or after the one selected. The substring of the same letter containing the selected letter is removed, and the string is shortened to remove the hole created. To solve the puzzle, the player has to remove all letters and obtain the empty string. If the player obtains a non-empty string in which no letter can be selected, then the player loses.

For example, if one starts with the string "ABBAABBAAB", selecting the first "B" gives "AAABBAAB". Next, selecting the last "A" gives "AAABBB". Selecting an "A" followed by a "B" gives the empty string. On the other hand, if one selects the third "B" first, the string "ABBAAAAB" is obtained. One may verify that regardless of the next selections, we obtain either the string "A" or the string "B" in which no letter can be selected. Thus, one must be careful in the sequence of selections chosen in order to solve a puzzle. Furthermore, there are some puzzles that cannot be solved regardless of the choice of selections. For example, "ABBAAAAB" is not a solvable puzzle.

Given a puzzle, your task is to determine whether it can be solved or not.

Input

Each case of input is specified by a single line. Each line contains a string of uppercase letters. Each string has at least one but no more than 150 characters. The input is terminated by the end of file.

Output

For each input case, print **solvable** on a single line if there is a sequence of selections that solves the puzzle. Otherwise, print **unsolvable** on a line.

Sample Input

```
ABBAABBAAB  
ABBAAAAB
```

Sample Output

```
solvable  
unsolvable
```