Problem H. Hard Refactoring

Input file: `hard.in`
Output file: `hard.out`

Helen had come upon a piece of code that uses a lot of "magical constants". She found a logical expression that checks if an integer x belongs to a certain set of ranges, like the one shown below:

```
x >= 5 && x <= 10 ||
x >= 7 && x <= 20 ||
x <= 2 ||
x >= 21 && x <= 25 ||
x >= 8 && x <= 10 ||
x >= 100
```

Helen does not like "magical constants", so she decided to *refactor* this expression and all similar ones in such a way, that the refactored expression still has the same Boolean result for all integers x, but it uses as few integer constants in its text as possible.

Integers in this problem, including integer x, come from the range of all signed 16 bit integers starting from -2^{15} ($-32\,768$) to $2^{15} - 1$ ($32\,767$) inclusive.

Input

The input file contains at most 1000 lines. Each line consists of either one comparison or two comparisons separated by *logical and* operator "`&&`". Each comparison starts with "`x`", followed by *greater-or-equals* operator "`>=`" or *less-or-equals* operator "`<=`", followed by an integer constant. When two comparisons are in the same line, the first one is always *greater-or-equals*, followed by *less-or-equals*.

All lines, but the last one, are terminated by *logical or* operator "`||`". All tokens in a line are separated by a single space and there are no trailing or leading spaces.

Output

Write the refactored expression to the output file in the same format as in the input. You can arrange lines in any order, as long as the resulting expression has the right format, produces the same Boolean result on all integers x, and contains the minimal possible number of integer constants in its text. Numbers must be formatted without leading zeros and there must be precisely one space between tokens on a line.

Write a single line with the word "`true`" if the expression is true on all integers. Write a single line with the word "`false`" if the expression is false on all integers.

Examples

hard.in	hard.out
x >= 5 && x <= 10 || x >= 7 && x <= 20 || x <= 2 || x >= 21 && x <= 25 || x >= 8 && x <= 10 || x >= 100	x <= 2 || x >= 5 && x <= 25 || x >= 100
x >= 10 && x <= 0	false
x <= 10 || x >= 0	true
x >= -32768	true