

## Problem H. Points and Lines

Input file: *standard input*  
Output file: *standard output*  
Time limit: 1 second  
Memory limit: 512 mebibytes

One day, you found an old scroll with strange texts on it.

You revealed that the text was actually an expression denoting the position of the hidden treasure. The expression consists of following three operations:

- From two points, yield a line on which the points lie.
- From a point and a line, yield a point that is symmetric to the given point with respect to the line.
- From two lines, yield a point that is the intersection of the lines.

The syntax of the expression is denoted by following BNF:

```
<expression> ::= <point>
<point> ::= <point-factor> | <line> "@" <line-factor> | <line> "@" <point-factor> | <point> "@" <line-factor>
<point-factor> ::= "(" <number> "," <number> ")" | "(" <point> ")"
<line> ::= <line-factor> | <point> "@" <point-factor>
<line-factor> ::= "(" <line> ")"
<number> ::= <zero-digit> | <positive-number> | <negative-number>
<positive-number> ::= <nonzero-digit> | <positive-number> <digit>
<negative-number> ::= "-" <positive-number>
<digit> ::= <zero-digit> | <nonzero-digit>
<zero-digit> ::= "0"
<nonzero-digit> ::= "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
```

Each `<point>` or `<point-factor>` denotes a point, whereas each `<line>` or `<line-factor>` denotes a line. The former notion of `<point-factor>`  $(X, Y)$  represents a point which has  $X$  for  $x$ -coordinate and  $Y$  for  $y$ -coordinate on the 2-dimensional plane. "@" indicates the operations on two operands. Since each operation is distinguishable from others by its operands' types (i.e. a point or a line), all of these operations are denoted by the same character "@". Note that "@" is left-associative, as can be seen from the BNF.

Your task is to determine where the treasure is placed.

### Input

The input consists of multiple datasets. Each dataset is a single line which contains an expression denoting the position of treasure.

It is guaranteed that each dataset satisfies the following conditions:

- The length of the string never exceeds  $10^2$ .
- If both operands of "@" are points, their distance is greater than 1.
- If both operands of "@" are lines, they are never parallel.
- The absolute values of points' coordinates never exceed  $10^2$  at any point of evaluation.

You can also assume that there are at most 100 datasets.

The input ends with a line that contains only a single "#".

### Output

For each dataset, print the  $X$  and  $Y$  coordinates of the point, denoted by the expression, in this order.

The output will be considered correct if its absolute or relative error is at most  $10^{-2}$ .

## Examples

standard input	standard output
((0,0)@(1,1))@((4,1)@(2,5))	3.00000000 3.00000000
((0,0)@(3,1))@((1,-3)@(2,-1))	3.00000000 1.00000000
(0,0)@(1,1)@(4,1)	1.00000000 4.00000000
(0,0)@((1,1)@(4,1))	0.00000000 2.00000000
((((0,0)@((10,20)@((30,40))))))	-10.00000000 10.00000000
((0,0)@(3,1))@((1,-3)@(2,-1))@(100,-100)@(100,100)	99.83681795 -91.92248853
#	