Assignment 1: Input-Output Specifications

(Professor: Dr. Ahmed; prepared by: Vera Kazakova)

Make sure your input follows the exact specifications below, down to the new lines, capitalization, and spelling.
Read in inputs in a loop with two choices:

char ‘e’: scan in and process an expression, print outputs, then scan next choice;
char ‘x’: exit program.

IMPORTANT: **exclude ALL PROMPTS FOR INPUT as well as any extraneous outputs**, such as a goodbye print on exit.
Below is the input/output clarification. Your program should expect to scan in what’s in red and print what’s in green.

e	← SCAN: just a single lower case character
<expression>	← SCAN: some expression in infix notation with balanced parenthesis
infix: <expression>	← PRINT: just reprint what you read in with “infix: “ at the front
parenthesis: balanced	← PRINT: check for balance in parens and print result
postfix: <converted expression>	← PRINT: process infix version and print postfix version
value: <number>	← PRINT: process postfix version and print value
e	← SCAN: just a single lower case character
<expression>	← SCAN: some expression in infix notation with unbalanced parenthesis
infix: <expression>	← PRINT: reprint this new read in expression
parenthesis: unbalanced	← PRINT: check for balance in parens; this example was not balanced
postfix: n/a	← PRINT: do not process into postfix, just print “n/a”
value: n/a	← PRINT: do not evaluate non-existent postfix, just print “n/a”
.	
.	← (keep scanning in ‘e’ followed by expressions, until you see ‘x’)
.	
x	← SCAN: just a single lower case character
	← EXIT: print nothing (not even a blank line) and just exit the program

If you test all sample inputs provided in the assignment, you should see exactly the following in your console:

```
e
(7 - 3) / (2 + 2))
infix: (7 - 3) / (2 + 2))
parenthesis: unbalanced
postfix: n/a
value: n/a
e
(5+6)*7-8*9
infix:  (5+6)*7-8*9
parenthesis: balanced
postfix: 5 6 + 7 * 8 9 * -
value: 5
e
(7 - 3) / (2 + 2)
infix: (7 - 3) / (2 + 2)
parenthesis: balanced
postfix: 7   3   -   2   2   +   /
value: 1
e
infix: 3+(4*5-(6/7^8)*9)*10,
parenthesis: balanced
postfix: 3 4 5 * 6 7 8 ^ /9 * - 10 * +
value: 203
e
1000 + 2000
infix: 1000 + 2000
parenthesis: balanced
postfix: 1000 2000 +
value: 3000
```