Exercise 36

Samantha Hafner

10/13/19

In Friedman and Wand's interpreter for their LET programming language, value-of is always passed an abstract syntax tree. The function is called at most one time from a non-recursive call in the execution of a program in the LET programming language.

Let the size of an abstract syntax tree be the integer number of expressions it represents. const-exp's and var-exp's are the smallest abstract syntax trees, with size 1, and each is evaluated with a single, and therefore finite number of calls to value-of. All other expressions, when evaluated by value-of result in either an error, terminating early with a finite number of calls to value-of, or in one or two recursive calls to value-of on a strictly smaller abstract syntax tree. By structural induction, all abstract syntax trees are evaluated with a finite number of recursive calls to value-of.

Therefore, the execution of any program in the LET programming language will terminate after a finite number of calls to the value of procedure.

Interpreter and specification for LET by Friedman and Wand.