Ph.D. Qualifying Exam (Spring 2002)
Programming Languages
Answer ALL questions

1. Lazy Evaluation (60 points)

Call-by-Need (Lazy Evaluation) is a parameter passing mechanism in which the evaluation of the actual parameter is deferred until the formal parameter is first used. When (and if) the evaluation occurs, the resulting value is saved for any subsequent evaluations of the formal parameter.

(a) How does this mechanism differ from call-by-value, call-by-reference, and call-by-name? Give examples to illustrate the difference.

(b) Describe how to implement lazy evaluation reasonably efficiently for the C language.

(c) Give a denotational semantics for call-by-value, call-by-reference, and call-by-need. You may build your solution by modifying the syntax and denotational semantics of the language in figures 7.1 and 7.2 of Schmidt's book. Describe only the modifications needed to include procedures and the various forms of procedure calls; you do not need to rewrite the whole denotational semantics.

2. Garbage Collection (40 points)

Describe the main advantanges and disadvantages of garbage collection. Why is it harder to garbage collect C than it is to garbage collect a language like Lisp? Sketch out the design of a garbage collector for the C language.