

2001 Paper 5 Question 6

Compiler Construction

Consider a language J which has

- Java-like syntax
- nested definitions of procedures within other procedures
- local variables (with static binding)
- raising and handling of named, parameterless exceptions

Explain a possible run-time data structure which a compiler for J might use.

[10 marks]

A naïve user of such a language suggests that the resultant compiled code will spend a significant fraction of execution time searching—both finding the store location corresponding to the use of a variable name and finding the exception handler corresponding to the raising of a given exception name.

Determine with justification whether this is so for your run-time data-structure proposed above.

[4 marks]

Now instead suppose a simple interpreter for J is written, so that searches for variable (or exception) names search the appropriate environment for the variable value or exception handler code. To what extent are these searches bounded by (a) the number of variables or exceptions in the program or (b) the dynamic or static nesting of procedures?

[6 marks]