

PROPOSED EXAM QUESTIONS FROM PLDI CONFERENCE READINGS

Group 1:

Code Compression

Adams, Lewis, Bjorkland, Melkote, Parks

1. How is wire code different from an executable file that has been gzipped?
2. What are the cases of problems that the Code Compression paper addresses?

Group 2

Fast, Effective Code Generation in a Just-in-Time Compiler

Yu, Richardson, Khosla, Perugini

1. When one uses JIT, we first produce byte code, compile it, and then run it. Why not compile the Java source code directly into native machine code?
2. What features of Java does the JIT exploit, and how does it exploit them, to make common sub-expression elimination faster?

Group 3:

Thin Locks: Featherweight Synchronization for Java

Bhushan, Rioux, Van Metre, Watkins

1. In Java, the garbage collector uses the third word of an object header to remove unused objects from memory and to move objects around in memory for the purpose of compaction. The garbage collector can run asynchronously or synchronously depending on the platform and the implementation of the Java Virtual Machine (JVM). Considering the implementation of thin locks, what are the effects that the implementation of thin locks have on the garbage collector?
2. In what cases would a thin lock be made into a fat lock and how would that be implemented.

Group 4:

Eliminating Array Bound Checking Through Dependent Types

Arrivillaga, Bussom, Grogan, Price

1. As presented, what is the difference between conservative and non-conservative array bounds checking? What, if any, are the trade-offs?
2. Relative to the focus of the paper, what is the definition and significance of a dependent type?