CSE 501: Implementation of Programming Languages

Goals:

- Understand how languages get implemented efficiently
- Understand how program analysis tools get built
- Understand what can and can’t be done, with how much effort
- Understand state of current research in efficient language implementation

Prerequisites:

- CSE 401 or equivalent
- CSE 505 or equivalent

Readings:

- main text: *Modern Compiler Implementation*, by Appel
- for reference: *Compilers: Principles, Techniques, and Tools*, by Aho, Sethi, and Ullman
- plus important papers from the literature

Grading:

• Midterm:	25%
• Final:	30%
• Homework:	20%
• Project:	25%

Course web page:

http://www.cs.washington.edu/education/courses/cse501/CurrentQtr

- office hours, office locations
- course mailing list signup instructions
- on-line copies of all lecture slides, handouts, assignments, etc.
- course project information and instructions

Rough Course Outline

Week	Topic
1/7	Intro; structure of compilers; standard optimizations Standard intermediate representations; control flow, data flow; dependence
1/14	Dataflow analysis; reaching constants, constant propagation ***NO CLASS ON REGULAR DAYS; make up class Monday afternoon?***
1/21	Lattice-theoretic data flow analysis framework; integer range analysis Data flow analyzer generators, frameworks
1/28	Advanced intermediate representations: def/use chains, control dependence tree, SSA form, VDG CSE; loop-invariant code motion
2/4	Inlining Interprocedural analysis; analysis with first-class functions, dynamically dispatched messages
2/11	Procedure specialization, partial evaluation Alias and pointer analysis
2/18	Register allocation Instruction scheduling
2/25	Garbage collection
3/4	Implementing functional and object-oriented languages
3/11	Dynamic (“JIT”) compilation