New course

6.893: Topics in Program Analysis

http://sdg.lcs.mit.edu/6.893/

Prof. Michael Ernst
mernst@lcs.mit.edu
http://sdg.lcs.mit.edu/~mernst/

Tentative meeting times: MW 1-2:30, 38-166
First meeting: Wed., Sept. 5, 1:00, 38-166

This 12-unit graduate seminar examines a variety of program analysis techniques. Topics include static analyses (as typically found in compilers and other programs that are provided source code), dynamic analysis (as found in profilers and other tools that work over program executions), and hybrid analyses that combine the two approaches. Sample analyses to be covered include model checking, theorem-proving, type inference and checking, dataflow analysis, concept analysis, profiling, test suite coverage, dynamic invariant detection, race detection, and visualization.

The focus of the course is the development and use of tools based on program analysis to enhance human capabilities and assist in program development tasks, including design, implementation, testing, documentation, and maintenance. (An alternative would be to concentrate on, say, optimization for the sake of faster execution, but people form the real bottleneck in software development.)

In addition to weekly readings of research papers and small exercises that provide experience with a variety of tools, the course includes a substantial team project performed in groups of 2-3. Sample projects will be provided, but students are also free to propose their own. Examples include proposing and evaluating a fundamental new technique; developing and assessing new algorithms to replace currently-used ones; translating a methodology to a new problem domain; evaluation of proposed techniques (for instance, via a user study); and applying known techniques to new problem domains, such as operating systems, networks, embedded systems, security, biology, aerospace, etc.

Prerequisite: 6.035 or instructor's permission.

The course awards Grad H and EC credit.