E0255 Jan 3:1

Compiler Design

Instructor

Y.N. Srikant
Email: srikant@iisc.ac.in

Teaching Assistant

Email:

Department: CSA

Course Time: Tue., Thu., 9.30 - 11 AM

Lecture venue: CSA 117

Detailed Course Page:

Announcements

Brief description of the course

This is an advanced course on compiler design and it assumes that the student is familiar with the topics of lexical analysis, parsing, semantic analysis, and intermediate code generation. The course will focus on optimization, code generation, and parallelization. Programming assignments will be based on LLVM.

Prerequisites

Basic topics in compiler design as covered in a UG course.

Syllabus

Control flow graphs and analysis; Dataflow analysis; Static single assignment (SSA); Compiler optimizations; Dependence analysis, Loop optimizations and transformations, Parallelization, Optimizations for cache locality, and Vectorization; Domain-specific languages, compilation, and optimization; Machine code generation; Register allocation, Instruction scheduling; Run time environment and storage management; Impact of language design and architecture evolution on compilers.

Course outcomes

The students will understand how advanced optimizations work in a compiler. They will also learn to program

optimizations and code generation using the LLVM framework.

Grading policy

50% for two programming projects, 30% for mid-term (two), 20% for final.

Assignments

Resources

1. A.V. Aho, M.S. Lam, R. Sethi, and J.D. Ullman, Compilers: Principles, Techniques, and Tools, 2nd ed., Addison-Wesley, 2007.

2. Y.N. Srikant and Priti Shankar, The Compiler Design Handbook, 2nd ed., CRC Press, 2008.

3. Current literature.