CENG444—Language Processors

Instructors: Cem Bozşahin. Aysun Bascetincelik (TA)


Topics: Assuming familiarity with formal grammars, i.e., CEng280 material, we will emphasize modern tools and techniques for compiler construction.

- Lexical analysis
  - Token recognition with FSMs
  - Automated tools for lexical analysis (e.g. antlr and lex)

- Syntactic analysis
  - Top-down parsing: recursive-descent
  - Predictive top-down parsing: (LL grammars)
  - Bottom-up parsing: deterministic shift-reduce parsing (LR grammars)
  - Attribute grammars
  - Abstract syntax and syntax trees
  - Parser generators: antlr (LL) & yacc (LR)

- Semantic analysis & code generation
  - Syntax-directed translation v. tree parsing
  - Type checking
  - Intermediate code and abstract/virtual machines
  - Run-time organization
  - Code generation

- Compiling non-imperative languages
  - Functions and lambda calculus
  - Combinators as primitives
  - G-machine as a Virtual Machine

Grading:

Project 20% (Phase 1—syntactic analysis and type checking)
20% (Phase 2—code generation and interpretation)
Mid-term 30% (in class, open book and notes)
Final 30% (open book and notes)

Notes:
1. No make-up exams; Let us know in advance if you are going to miss an exam.
2. Exams will cover portions of the text and the material covered in class. It is your responsibility to keep track of announcements (handouts, project, etc.). Class attendance is expected; we usually have lively discussion about compiling technology in the world—from ironing machines and refrigerators to Java, C, etc., and of course the project topic.
3. Read the metu.ceng.course.444 newsgroup regularly for course/project announcements and off-class discussion.