CS 432 Midterm 2 Study Guide
Fall 2019

You are expected to be able to perform the following tasks on the exam:

• Compare and contrast top-down vs. bottom-up parsing.

• Build a shift-reduce parser for an SLR(1) grammar.
  – Construct LR(0) item sets and automata.
  – Construct SLR(1) parsing tables (ACTION and GOTO).

• Discuss various static analysis considerations relevant to compilers.
  – Describe the visitor design pattern and how it can help during the construction of a compiler.
  – Explain the benefits and costs of static and dynamic type checking.

• Discuss various architecture and runtime system considerations relevant to compilers.
  – Instructions, opcodes, operands, and assembly/machine code.
  – Standard linkage conventions and how the stack is used to enable procedure calls.
  – Heap allocation and deallocation strategies.

• Analyze and generate code for Decaf programs.
  – Construct symbol tables.
  – Perform type inference and type checking.
  – Derive type safety proofs.
  – Convert Decaf code into equivalent ILOC in static single-assignment form with the calling conventions discussed in class

• Define and discuss specific terms or vocabulary related to any of the above concepts, including a detailed description of why and how they are relevant to the construction of a compiler.