Compilers [Spring 2025]
Test I

NAME: _____

Instructions:

1) This test is 7 pages in length.

2) You have 75 minutes to complete and turn in this test.

3) Short-answer and essay questions will be graded on how clearly you have communicated the necessary ideas. Respond in complete English sentences. Avoid using bullet points and enumerated lists. Respond at the level of detail discussed in class.

4) All grammars must contain (a) no circular rules of the form N::=N, (b) no syntactic categories lacking a base case (i.e., every syntactic category must generate at least one string), (c) no repeated rules, and (d) exactly one production for the starting nonterminal S having the form S::=N$.

5) Use the same notations, assumptions, invariants, etc. that we have been using in class.

6) This test is closed everything, including books, notes, papers, computers, phones, laptops, smartwatches, smartglasses, AI, LLMs, friends, neighbors, etc. Do not talk to another student during the test. Do not look at another student's answers during the test.

1. [4 points] What is a language? What is a programming language? [2 sentences]

2. [11 points] Write flex-style REs for IDs and natural-number literals in DJ.

3. [15 points] For all strings s1 and s2, s1 is a substring of s2 when there exist (possibly empty) strings s3 and s4 such that s2=s3s1s4. Now, write an RE and a minimum-state DFA matching all strings over {0,1} that contain neither 00 nor 11 as a substring.

4. [15 points] [Essay] Explain the format of .l and .y files.

5. [10 points] Define a CFG G, prove G∈LL(1)∩LR(0), and show an LR(0) parse trace of a valid input.

6. [12 points] Define a CFG G, prove G∈LR(0)\LL(1), and show an LR(0) parse trace of an invalid input. The \ symbol here means "set difference", so G must be in LR(0) but not LL(1), which is sometimes written G∈LR(0)-LL(1).

7. [18 points]
Define a CFG G, prove G∈ (LL(1)∩SLR)\LR(0), and show pseudocode implementing a predictive parser for G.

8. [15 points]
Define a CFG G, prove G∈SLR\LL(1)\LR(0), show an LR(0) parse trace of a valid input that ends in a conflict, and show an SLR parse trace of that same valid input.

Undergraduates stop here. The remaining problem is for graduate students.

9. [10 points] Identify and, at a high level, prove the asymptotic worst-case running time of the algorithm we defined in class, for converting an arbitrary RE into an NFA. Focus on the main ideas; do not go through the details.