Compilers [Spring 2023]
Test II

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've 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) This test is closed books, notes, papers, friends, phones, neighbors, smartwatches, etc.

5) Use the backs of pages in this test packet for scratch work. If you write more than a final answer in the area next to a question, circle your final answer.

6) Write and sign the following: "I pledge my Honor that I have not cheated, and will not cheat, on this test."

____

____

Signed: ______________________________________

1. [3 points] [1 sentence] In DJ, what are class members?

2. [5 points] [1-2 sentences]
What is the Curry-Howard isomorphism? Hit the main points from class.

3. [5 points] [1-3 sentences]
Using big-O notation, what are the worst-case running times of all the parsing algorithms discussed in class?

4. [5 points]
Implement the join function for djc, as discussed in class.

5. [10 points] Draw a minimum-state DFA recognizing all base-3 numbers that have a remainder of 1 or 3 when divided by 4.

6. [20 points] [Essay] Describe the symbol-table data structures used in djc.

7. [15 points]

Recall that context-free languages are those that can be specified by a CFG and, equivalently, can be recognized by a pushdown automaton (i.e., “an NFA with a stack”). Write high-level explanations that context-free languages are closed under (a) union and (b) concatenation but are not closed under (c) intersection and (d) complement.

Hint: At some point you may wish to consider $L_1=\{0^a1^a2^b\}$ and $L_2=\{0^a1^b2^b\}$ where $a,b\geq 0$.

8. [37 points] [The following page is blank, to provide additional space to respond.] Recall that LALR parsers may act, at each step, by shifting, reducing, accepting, or signaling an error. Hence, there are 6 possible pairings of non-error actions: shift-shift, shift-reduce, shift-accept, reduce-reduce, reduce-accept, and accept-accept. For each of these 6 possible pairings, explain whether an LALR parser could have such a conflict. If it could not have such a conflict, briefly explain why. If it could have such a conflict, (a) define an example grammar G that causes an LALR parser to have the conflict, (b) show enough of G's LALR DFA to pinpoint the conflict, (c) define another grammar H equivalent to G, (d) show that H's LALR parse table is conflict free, and (e) show an LALR parse trace of a valid input according to H. For this problem, grammars may not contain a repeated rule and may not contain a rule of the form N ::= N.

[This page provides additional space for Problem 8.]

Undergraduates stop here. The remaining problems are for graduate students.

9. [6 points] Person P implemented 1023 compilers, each with a front and back end. Every "end" P has written contributes to a complete compiler. What are the minimum and maximum numbers of "ends" P has written?

10. [6 points]

Draw an LL(2) parse table for CFG A, which has the rules S::=Q$, Q::=XQXQ, Q::=ε, and X::=z. If there's a conflict, rewrite A into an equivalent CFG B and show that B has no conflicts in its LL(1) table.