

CS4120/4121/5120/5121—Spring 2019

Homework 2

Syntactic Analysis

Due: Friday, February 8, 11:59PM

0 Instructions

0.1 Partners

You may work alone or with *one* partner on this assignment. But remember that the course staff is happy to help with problems you run into. Use Piazza for questions, attend office hours, or set up meetings with any course staff member for help.

0.2 Homework structure

There are two parts of the homework. The first part is required of all students. The second part is required of students taking CS5120, but those enrolled in CS4120 are welcome to try it for good **HARMA**.

0.3 Tips

You may find the Dot and Graphviz packages helpful for drawing graphs. You can get these packages for multiple OSes from the [Graphviz download page](#).

1 Problems

1. Context-free grammars

Consider the following BNF grammar over the alphabet $\{t, n, ::=, +\}$, where the start symbol is P :

$$P \rightarrow A \mid PA$$

$$A \rightarrow n ::= R$$

$$R \rightarrow S \mid R+S$$

$$S \rightarrow \varepsilon \mid E$$

$$E \rightarrow t \mid n \mid tE \mid nE$$

- Briefly identify two reasons why this grammar is not LL(1).
- Is the string $n ::= +tn$ in the language of the grammar? If so, show the rightmost derivation of the string. If not, explain why it cannot be derived.
- Show the start state for the LR(1) automaton for this grammar. (Hint: remember to take the closure of items.)

2. Ambiguous grammars

Consider the following grammar:

$$\begin{aligned} E &\rightarrow E Op E \mid (E) \mid \langle \text{num} \rangle \\ Op &\rightarrow + \mid * \mid ^ \end{aligned}$$

This grammar is ambiguous. We would like a grammar to derive parse trees in which exponentiation (^) has higher precedence than multiplication (*), and both have higher precedence than addition (+). Exponentiation should be right-associative, while multiplication and addition should be left-associative.

- Give an example to show that the grammar is ambiguous.
- Write an LL(1) grammar that accepts the same string as this grammar and respects the desired operator precedence. Associativity need not be enforced, however.
- Show the derivation of the expression $2^3 * 4 + 5$ using your grammar in part (b).
- Write an LR(1) grammar that accepts the same language, but enforces both the precedence and associativity of the operators. You need not show the parsing tables.

3. LR grammars

Is the following grammar LALR(1)? What about LR(1)?

$$\begin{aligned} S &\rightarrow A \\ A &\rightarrow aBb \\ B &\rightarrow b \\ B &\rightarrow \varepsilon \end{aligned}$$

2 Problem for CS5120

4. Predictive parsing

Give an unambiguous grammar that is not LL(1), but for which every nonterminal has a unique applicable production given the first token, i.e., for each nonterminal, the FIRST sets of its productions are disjoint.

3 Submission

Submit your solution as a PDF file on CMS. This file should contain your name, your NetID, all known issues you have with your solution, and the names of anyone you have discussed the homework with.