CS412 Homework 2 Due: 20 February 2006

Write your answers *very neatly*. Turn in your assignment in class on the due date.

1. Consider the following grammar:

$$\begin{array}{lcl} S & \rightarrow & A\ a \\ A & \rightarrow & d\ a\ b \mid B\ b \\ B & \rightarrow & c\ A \mid S\ c \end{array}$$

   (a) Compute the nullable, FIRST, and FOLLOW sets.

   (b) Construct the LL(1) parsing table and explain why the grammar is not LL(1).

   (c) Write an LL(1) grammar that accepts the same language. Give evidence that your grammar is LL(1).

2. The following grammar describes the language of regular expressions:

$$R \rightarrow R\ \mathit{bar}\ R \mid R\ R \mid R\ \mathit{star} \mid (\ R\ ) \mid \epsilon \mid \mathit{letter}$$

   where *bar*, *star*, *letter*, '(', and ')' are all terminals. This is an ambiguous grammar. We would like to write an unambiguous grammar for the same language, such that the Kleene star operation has higher precedence than concatenation; and, in turn, concatenation has higher precedence than alternation.

   (a) Write an LL(1) grammar which accepts the same language and has the desired operator precedence. Show the LL(1) parsing table for this grammar.

   (b) Write an LR(1) grammar which accepts the same language, respects the desired operator precedence, and is such that alternation is left-associative, but concatenation is right-associative.

   (c) Write the parse tree for the expression `a | b c * d | e` using the LR(1) grammar;

3. Construct the LR(0) automaton for the following grammar, and then use it to determine whether the grammar is SLR:

$$\begin{array}{lcl} S & \rightarrow & B\ \$ \\ B & \rightarrow & id\ P \mid id\ (\ E\ ] \\ P & \rightarrow & \epsilon \mid (\ E\ ) \\ E & \rightarrow & B \mid B, E \end{array}$$

4. Consider a robot arm that accepts two commands: $\downarrow$ puts an apple in a bag, and $\uparrow$ takes an apple from the bag. Assume that the robot arm starts with an empty bag. A valid command for the robot arm should have no prefix that takes more apples than it puts in the bag. For instance, $\downarrow\downarrow\uparrow\uparrow$ and $\downarrow\uparrow\downarrow$ are valid command, but $\downarrow\uparrow\uparrow\downarrow\downarrow$ is not.

(a) Write an LR(1) grammar that accepts this language.

(b) Build the parsing table to show that your grammar is LR(1).

(c) Show the parsing stack and the actions that the LR(1) parser performs for the input string $\downarrow\downarrow\uparrow\uparrow\downarrow$.

(d) Is your grammar LR(0)? Justify your answer.

5. Compare the LL(1) and LALR(1) parsing techniques, indicating their advantages and disadvantages. Develop a set of criteria for comparing parsing techniques and compare the two parsing methods with respect to your criteria. Assume that automated tools are available for each technique.