CPSC 411: Fall Midterm Exam, 7 March 2006

Robin Cockett

March 6, 2006

This exam is worth 25% of the credit for this course. There are 50 marks available: answer all the questions below.

1. (10 points)

   Describe the following:

   (a) The main stages of compiling a program: be careful to describe the structure of the inputs and outputs, the complexity of each step, and the stages which can generate errors.

   (b) The difference between a syntax tree and a parse tree.

   (c) Where context free grammars are used and why one (generally) wants the grammar to be unambiguous.

   (d) The complexity of determining whether a grammar is unambiguous.

   (e) Why a programming language designer might want to describe a language as an LL(1) grammar.

2. (5 points)

   Draw a minimal deterministic finite automaton which recognizes the same set of strings as those determined by the regular expression $a((b^*c)|a)^*$.

3. (20 points)

   Given the following grammar:

```
Exp -> Term + Exp | Term | let Decls in Exp end.
Term -> Factor * Term | Factor.
Factor -> id | num.
Decls -> Decls Decl  | .
Decl -> id = Exp ;.
```

   (a) Give three examples of strings recognized by this grammar.

   (b) Calculate the vital statistics of the grammar: that is

      i. Which non-terminals are nullable?

      ii. What are the first sets of each nonterminal?

      iii. What are the follow sets of each nonterminal?

      iv. Which non-terminals are endable?

      v. Which nonterminals are left recursive?

   (c) Draw the the LL(1) parsing table for this grammar and explain why this grammar is not LL(1).

   (d) Transform this grammar to make it LL(1): explain the steps you take. How do you know the resulting grammar is LL(1)?

4. (15 points)

   Consider the following grammar for boolean expressions:

```
Disj -> Disj or Conj | bool and Conj | bool.
Conj -> Conj and bool | bool.
```

   Explain the answers to the following questions:

   (a) Is the grammar LL(1)?

   (b) Is this grammar LR(0)?

   (c) Is this grammar SLR(1)?

   Finally draw the SLR(1) parsing table for the grammar.