Second Semester Examinations 2010/11

Decision, Computation and Language

TIME ALLOWED : Two Hours

INSTRUCTIONS TO CANDIDATES

Answer **ALL** questions from Section A and **TWO** from Section B.

If you attempt to answer more questions than the required number of questions (in any section), the marks awarded for the excess questions answered will be discarded (starting with your lowest mark).

Section A

1. Consider the following **non-deterministic** finite automaton, having three states $\{q_1, q_2, q_3\}$, where q_1 is the initial state, and q_1 is also the (only) accepting state. The transitions (using alphabet $\{\mathtt{a}, \mathtt{b}\}$) are the following.

$$\phi(q_1, \mathtt{a}) = \{q_1, q_2\}; \;\; \phi(q_1, \mathtt{b}) = q_2; \;\; \phi(q_2, \mathtt{a}) = q_3; \;\; \phi(q_3, \mathtt{a}) = q_1$$

a. Write down three strings that are accepted by this automaton. [3 marks]

b. Give an informal English-language description of the kinds of strings that are accepted by the automaton. [3 marks]

c. Write down either a regular expression or a **deterministic** finite automaton that accepts the same language. [3 marks]

d. Describe a general method for converting a **non-deterministic** finite automaton to a **deterministic** finite automaton that accepts the same language. [6 marks]

2. a. Write down a statement of the Pumping Lemma for Regular Languages. In a finite automaton, what would be the special feature of the transitions that correspond to part of a string that can be "pumped"? [5 marks]

b. Consider the language L over alphabet $\{\mathtt{0}, \mathtt{1}\}$, defined by the following set of words:

$$L \; = \; \{ \; \mathtt{0}^n \mathtt{1}^m \mathtt{0}^{n-m} \; : \; n \; > \; m \; \geq \; 0 \, \}$$

Using the Pumping Lemma for Regular Languages, prove that L is not a regular language.

[10 marks]

3. Give examples of each of the following. (Full credit requires a formal definition of a language with the relevant properties, but not a proof.)

a. A **deterministic context-free** language that is **not** a **regular** language. [3 marks]

b. A **context–free** language that is **not** a **deterministic** context-free language. [4 marks]

c. A **recursive** language that is **not** a **context–free** language. [4 marks]

d. A **recursively enumerable** language that is **not** a **recursive** language. [4 marks]

4. Consider the following context-free grammar. $(\{S, A, B, C\}, \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}, P, S)$ where P is the set of rules:

$$\begin{array}{lcl} S & \longrightarrow & ABCS \mid \mathtt{d} \\ A & \longrightarrow & \mathtt{a}A \mid \epsilon \\ B & \longrightarrow & \mathtt{b}B \mid \epsilon \\ C & \longrightarrow & \mathtt{c} \end{array}$$

a. Write down a LL(1) parse table for the grammar. [7 marks]

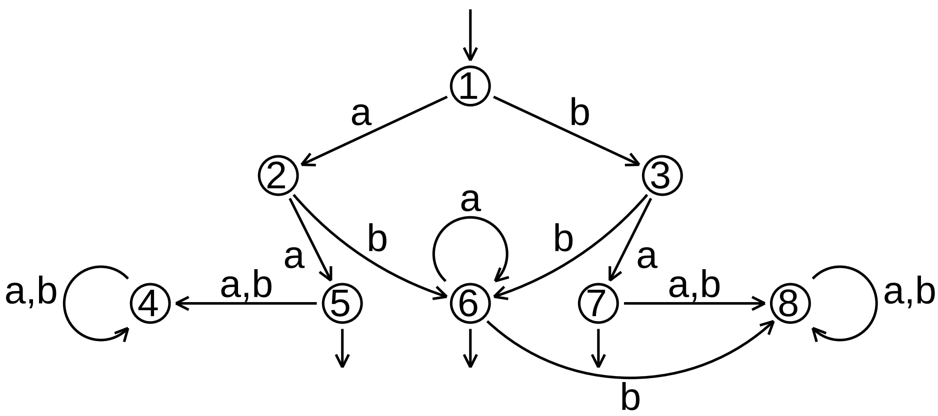
b. What is the FIRST set of symbol S? Which symbols are nullable? [3 marks]

c. Write down a derivation (using the grammar) of the word `bbcacd`. [5 marks]

Section B

Answer **two** questions in this section.

1. **a.** In the DFA shown below, identify which states are inaccessible and which are indistinguishable. (You may use the standard algorithms or some other approach.) Describe in general how the information about which states of a DFA are inaccessible/indistinguishable may be used to construct a minimal equivalent DFA. Hence write down a minimal DFA equivalent to the one below. [15 marks]



b. Prove that if a DFA has no inaccessible or indistinguishable states, then it does in fact have a minimal number of states. [10 marks]

2. **a.** Describe a deterministic pushdown automaton that accepts the set of all words over alphabet $\{\mathtt{a}, \mathtt{b}\}$ of the form $\mathtt{a}^n\mathtt{b}\mathtt{a}^m\mathtt{b}$ where $m \geq n$. [10 marks]

b. Now consider the language consisting of palindromes (words that are unchanged by reversing the order of their letters) over the alphabet $\{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$.

(i) Can this language be accepted by a deterministic pushdown automaton? (explain your answer). [8 marks]

(ii) Write down a Greibach normal form grammar for the language. [7 marks]

3. **a.** Give an unrestricted grammar that generates the language consisting of all words of the form $\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n$ (where $n = 0, 1, 2, \ldots$). Show how your grammar derives the word `aabbcc`. [15 marks]

b. Define *recursive* and *recursively enumerable*. Explain why a language L is recursive if and only L and its complement are recursively enumerable. [10 marks]