VICTORIA UNIVERSITY OF WELLINGTON

Te Whare Wananga o te Upoko o te Ika a Maui

EXAMINATIONS — 2004

END-YEAR

COMP 202

Formal Methods of Computer Science

Time Allowed: 3 Hours

Instructions: There are seven (7) questions, worth fourteen (14) marks each, making ninety-eight (98) marks in total.
Answer all the questions.

You may use printed foreign language dictionaries.
You may not use calculators or electronic dictionaries.

Question 1.

Let e_1 and e_2 be integer-valued expressions, and x be a variable of type integer. Below are semantic functions for a fragment of the language of while-programs.

$$\begin{array}{rcl}
\mathcal{V}(e_1 + e_2, \mathcal{S}) & = & \mathcal{V}(e_1, \mathcal{S}) + \mathcal{V}(e_2, \mathcal{S}) \\
\mathcal{V}(e_1 - e_2, \mathcal{S}) & = & \mathcal{V}(e_1, \mathcal{S}) - \mathcal{V}(e_2, \mathcal{S}) \\
\mathcal{V}(n, \mathcal{S}) & = & n \\
\mathcal{V}(x, \mathcal{S}) & = & \mathcal{S}(x) \\
\mathcal{M}(x := e_1, \mathcal{S}) & = & \mathit{update}(\mathcal{S}, x, \mathcal{V}(e_1, \mathcal{S}))
\end{array}$$

(a) Briefly describe each of the following:

1. $\mathcal{S}$
2. $\mathcal{V}$
3. $\mathcal{M}$
4. update [6 marks]

(b) We wish to add operations on *sets of integers* to the language, as follows. Let f_1 and f_2 be set-valued expressions, and let y be a variable of type set.

$f_1 \cap f_2$	is a set-valued expression whose value is the intersection of f_1 and f_2
$f_1 \cup f_2$	is a set-valued expression whose value is the union of f_1 and f_2
$\mathit{mkset}(e_1)$	is a set-valued expression whose value is the set containing just e_1

Extend the definition of $\mathcal{V}$ above to incorporate set-valued expressions and variables. [8 marks]

Question 2.

(a) Describe the main difference between non-deterministic finite automaton (NFA) and a deterministic finite automaton (FA). [2 marks]

(b) Describe the main idea behind the construction of an FA from an NFA, such that both machines define the same language. [4 marks]

(c) Let $M_1 = (Q, \Sigma, \delta, q_0, F)$, where:

$$\begin{array}{rcl} Q & = & \{S_1, S_2, S_3, S_4\} \\ \Sigma & = & \{a, b\} \\ q_0 & = & S_1 \\ F & = & \{S_4\} \end{array}$$

δ	a	b
S_1	$\{S_2, S_3, S_4\}$	$\{\}$
S_2	$\{S_3, S_4\}$	$\{\}$
S_3	$\{S_4\}$	$\{\}$
S_4	$\{S_4\}$	$\{S_1, S_2, S_3\}$

(i) Draw M_1. [3 marks]

(ii) Find an FA which accepts the same language as M_1. [5 marks]

Question 3.

(a) State what it means for a finite automaton to accept a string. [4 marks]

(b) The pumping lemma states that if L is a regular language then there is a number p, such that $(\forall s \in L)(|s| \geq p \Rightarrow s = xyz)$, where:

1. $(\forall i \geq 0)xy^iz \in L$
2. $|y| > 0$
3. $|xy| \leq p$

(i) Outline the proof of the pumping lemma. [5 marks]

(ii) Show that the language $L_1 = \{a^nb^n | n \in \mathbb{N}\}$ is not a regular language. [5 marks]

Question 4.

The following context-free grammar generates the language $L_1 = \{a^i b^j c | j \leq i \leq 2j\}$:

$$\begin{aligned} S_1 &\to T_1 c \\ T_1 &\to aT_1 b \quad | \quad aaT_1 b \quad | \quad \Lambda \end{aligned}$$

(a) State Σ, N, and S for the above grammar. [1 mark]

(b) List the **nullable nonterminals**. [1 mark]

(c) Find an equivalent grammar with no Lambda productions. [3 marks]

(d) Give two different **leftmost derivations** of the string $aaabbc$. [2 marks]

(e) Find an equivalent **unambiguous grammar**. [2 marks]

(f) Define a regular grammar for the language $L_2 = \{(ab)^i | i \geq 0\}$ [3 marks]

(g) Hence, define a context-free grammar for $L_1 \cup L_2$. [2 marks]

Question 5.

Consider the following context-free grammar:

$$
\begin{aligned}
S &\rightarrow XaY \\
X &\rightarrow aY \mid aZ \\
Y &\rightarrow c \mid d \\
Z &\rightarrow d \mid \Lambda
\end{aligned}
$$

(a) Explain what it means for a grammar to be in LL(1) form. [2 marks]

(b) Compute:

1. $\mathit{first}(X)$
2. $\mathit{follow}(X)$
3. $\mathit{first}(Y)$
4. $\mathit{first}(Z)$
5. $\mathit{follow}(Z)$ [3 marks]

(c) Show that the above grammar is **not** in LL(1) form. [2 marks]

(d) Rewrite the grammar so that it is in LL(1) form. [3 marks]

(e) This is the heading for a procedure $ParseS$ for a recursive-descent parser to recognize the S nonterminal:

procedure $ParseS$ (**in out** ss)
begin
 ...
end

The parameter ss, providing the input for the parser, is a sequence of terminal symbols.

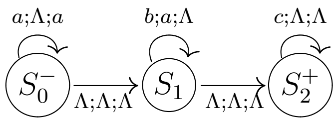
Complete the $ParseS$ procedure. You do not need to return a parse tree. You may assume that procedures $ParseX$, $ParseY$ and $ParseZ$ for recognizing the other nonterminals are already written. [4 marks]

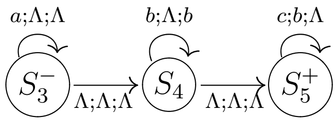
Question 6.

(a) Define pushdown automata that accept each of the following languages:

1. The language described by the regular expression $\mathbf{ab^*c}$ [2 marks]
2. The language $\{a^m b^m | m \geq 1\}$ [2 marks]
3. The language consisting of strings over the alphabet $\{a, b\}$ with the same number of as and bs, but occurring in any order [2 marks]

(b) The following PDAs P_1 and P_2 define context-free languages L_1 and L_2 (respectively):

P_1 :


P_2 :


1. Describe the languages L_1 and L_2. [2 marks]
2. Is the concatenation (product) $L_3 = L_1 L_2$ context-free?
   If so, draw a PDA for L_3. If not, explain why not. [3 marks]
3. Is the intersection $L_4 = L_1 \cap L_2$ context-free?
   If so, draw a PDA for L_4. If not, explain why not. [3 marks]

Question 7.

Let T be a Turing machine with input alphabet Σ.

(a) Define the three sets:

1. $\mathrm{accept}(T)$
2. $\mathrm{reject}(T)$
3. $\mathrm{loop}(T)$ [3 marks]

(b) Hence define:

1. Computable language
2. Computably enumerable language [4 marks]

(c) A compiler is a program which translates programs written in one language (the source language) to programs written in another (the target language), such that executing the target program on some input is the same as executing the source program on the same input.

For each of the following statements, state whether it is true or false, and why:

1. Java programs can be compiled to Turing machines;
2. Java programs can be compiled to PDAs;
3. PDAs can be compiled to Turing machines;
4. It is possible to write a program which can take any Java program and its input and correctly state whether the Java program will halt on that input. [7 marks]

****