TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI
VICTORIA
UNIVERSITY OF WELLINGTON

EXAMINATIONS — 2007

MID-TERM TEST

COMP/SWEN 202
Formal Foundations of Computer Science and Software Engineering

Time Allowed: 90 minutes

Instructions: There are **four** (4) questions.
Answer **all** the questions.
Show **all** your working.

Question 1. [15 marks]

For each the following languages described below, **(i)** write a regular expression that defines the language, and **(ii)** draw a transition diagram for a (nondeterministic) finite acceptor that recognises the language:

(a) The set of all strings over $\{a, b, c\}$ containing at least one a and at least one b.

(b) The set of all strings over $\{a, b, c\}$ in which every occurrence of a is immediately followed by a c, and no occurrence of c is immediately followed by an a.

Question 2. [15 marks]

Consider the NFA $M = (Q, q_I, A, N, F)$, where:

- $Q = \{1, 2, 3, 4, 5, 6\}$
- $q_I = 1$
- $A = \{a, b\}$
- $N(1, a) = \{2, 3\}$,
  $N(2, a) = \{2, 4\}$,
  $N(2, b_)) = \{5\}$,
  $N(3, b) = \{3, 6\}$,
  $N(1, x) = \{\}$, otherwise
- $F = \{4, 6\}$

(a) Draw a transition diagram for M.

(b) Describe, in English, the language recognised by M.

(c) Draw a transition diagram for a complete DFA equivalent to M.

(d) Write a regular expression which defines the language recognised by M.

Question 3. [20 marks]

Let M_1 and M_2 be two NFAs, where $M_i = (Q_i, q_{I_i}, A_i, N_i, F_i)$ for $i = 1, 2$.

(a) Explain, in English, how M_1 and M_2 can be combined to obtain an NFA that recognises $L_1 \cup L_2$, where L_1 and L_2 are the languages recognised by M_1 and M_2, respectively.

(b) Give a mathematical definition of the NFA described in part (a), and give a brief argument explaining why this NFA recognises $L_1 \cup L_2$.

(c) Draw a transition diagram for the NFA obtained by applying this construction to the two NFAs you drew for Question **1**.

Question 4. [20 marks]

Consider the following grammar:

$$\begin{aligned} E &\rightarrow F \mid F * F \mid F + F \\ F &\rightarrow 1 \mid 2 \mid 3 \mid (\ E - E\ ) \end{aligned}$$

(a) Draw a parse tree for the string $1 + 2 * 3$.

(b) Draw a parse tree for the string $1 + (2 - 1 + 3)$.

(c) Explain, giving an example, why this grammar is ambiguous.

(d) Show how you would construct an equivalent unamibiguous grammar by treating $*$ and $+$ as *left associative*.

(e) Show how you would construct an equivalent unamibiguous grammar by treating $*$ as having *higher precedence* (i.e. as binding more tightly) than $+$.

In parts (d) and (e), you should give the new grammar, explain how and why you have changed it, and show how the new grammar addresses the example you used in part (c).

****