TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI

VICTORIA

UNIVERSITY OF WELLINGTON

EXAMINATIONS — 2007

END-YEAR

COMP 202
Formal Methods of Computer Science

Time Allowed: 3 Hours

Instructions: There are nine (9) questions, worth varying numbers of marks.
There are one hundred and eighty (180) marks in total.
Answer all the questions.

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

L_0, a context-free language

The following describes a context-free language L_0, which is a small fragment of English. Use it in answering questions 1–3.

You may wish to tear this page from the front of the paper.

The sentences of L_0 are defined by a context free grammar G_0.

The terminal symbols of G_0 are English words, identified by single lower-case letters, possibly with subscripts:

terminal	word
a	`a`
t	`ten`
c_1	`cat`
c_2	`cats`
r_1	`rat`
r_2	`rats`
e_1	`eats`
e_2	`eat`

The nonterminal symbols of G_0 are single capital letters, possibly with subscripts. The nonterminals stand for parts of English sentences, as follows:

nonterminal	sentence part
S	sentence
P_n	noun phrase
P_v	verb phrase
N	noun
N_1	singular noun
N_2	plural noun
V	verb
D	determiner

The productions of G_0, labelled $P_1..P_{15}$, are as follows:

$$
\begin{array}{llll}
S & \rightarrow & P_n\ P_v & (P_1, P_2) \\
P_n & \rightarrow & D\ N & (P_3) \\
P_v & \rightarrow & V \mid V\ P_n & (P_4, P_5) \\
N & \rightarrow & N_1 \mid N_2 & (P_6, P_7) \\
N_1 & \rightarrow & c_1 \mid r_1 & (P_8, P_9) \\
N_2 & \rightarrow & c_2 \mid r_2 & (P_{10}, P_{11}) \\
V & \rightarrow & e_1 \mid e_2 & (P_{12}, P_{13}) \\
D & \rightarrow & a \mid t & (P_{14}, P_{15})
\end{array}
$$

Question 1. 26 marks

Refer to the definition of L_0 on page 2 of the paper in answering this question.

(a) State Σ, N, S, and P for G_0. Use the labels $P_1..P_{15}$ as names for the productions. [6 marks]

(b) Using G_0, give a derivation for each of the following sentences:

(i) `a cat eats a rat`

(ii) `ten cats eat`

(iii) `a rat eats ten cats`

(iv) `ten cats eats a rats` (*) [12 marks]

(c) The sentence marked (*) above is not valid English: English requires agreement in number (singular/plural) between determiners, nouns, and verbs.

(i) Define the term "covering grammar".

(ii) Briefly describe two techniques that may be used to define a recogniser for the fragment of English that uses the vocabulary of G_0 but that correctly observes number agreement. [8 marks]

Question 2. 26 marks

(a) State the two requirements for a context-free grammar to be LL(1). [6 marks]

(b) Using G_0, as defined on page 2 of this paper, compute the following.

(i) $\mathit{first}(N_1)$

(ii) $\mathit{first}(N_2)$

(iii) $\mathit{first}(N)$

(iv) $\mathit{first}(V)$

(v) $\mathit{first}(P_v)$ [10 marks]

(c) Show that G_0 is not LL(1). [10 marks]

Question 3. 24 marks

Refer to the definition of L_0 on page 2 of the paper in answering this question.

Let G_1 be the grammar obtained from G_0 by adding a new nonterminal P_o (object phrase), removing productions P_4 and P_5, and adding the following productions:

$$
\begin{array}{lcll}
P_v & \rightarrow & V\ P_o & (P_{16}) \\
P_o & \rightarrow & \Lambda \mid P_n & (P_{17}, P_{18})
\end{array}
$$

(a) Using G_1, compute:

(i) $\mathit{first}(P_o)$

(ii) $\mathit{follow}(P_o)$ [6 marks]

(b) Show that G_1 is LL(1). [6 marks]

(c) The following are two procedures of a recursive descent parser for L_0. The global variable ss initially holds the sequence of terminal symbols that is to be parsed.

```
procedure N
   if empty(ss) then
      error
   else if head(ss) = c1 or head(ss) = r1 then
      N1
   else
      N2
```

```
procedure N1
   if empty(ss) then
      error
   else if head(ss) = c1 then
      ss := tail(ss)
   else if head(ss) = r1 then
      ss := tail(ss)
   else
      error
```

Using the definitions of N and N_1 as a model, define procedures for:

(i) V

(ii) P_o

(iii) S

[12 marks]

Question 4. 16 marks

(a) Regular expressions over an alphabet Σ may be defined inductively as follows:

- $\emptyset$ and $\mathbf{\Lambda}$ are regular expressions
- for every $x \in \Sigma$, $\mathbf{x}$ is a regular expression
- if $\mathbf{r}$ and $\mathbf{s}$ are regular expressions, so are $\mathbf{r}^*$, $\mathbf{rs}$, and $\mathbf{r} + \mathbf{s}$.

Let $\mathbf{r_2}$ be a regular expression describing some language L_2. Describe how to construct the transition diagram for a nondeterministic finite automaton with Λ transitions (NFAΛ) that accepts L_2. [8 marks]

(b) Let L_3 be the language accepted by F_3, a deterministic finite automaton (FA) defined by

$$F_3 = (\Sigma, Q, q_0, F, \delta).$$

Give a *formal* proof that there is a nondeterministic finite automaton (NFA)

$$N_3 = (\Sigma', Q', q_0', F', \delta')$$

which accepts L_3. [8 marks]

Question 5. 16 marks

(a) Let L_4 be the language accepted by M_4, a nondeterministic finite automaton with Λ transitions (NFAΛ) defined by
$M_4 = (Q, \Sigma, \delta, q_0, F)$, where:

- $Q = \{S_0, S_1, S_2\}$
- $\Sigma = \{a, b\}$
- $q_0 = S_0$
- $F = \{S_2\}$

and δ is given by the table:

δ	a	b	Λ
S_0	$\{S_0\}$	$\{\}$	$\{S_1\}$
S_1	$\{\}$	$\{S_1\}$	$\{S_2\}$
S_2	$\{S_2\}$	$\{\}$	$\{S_2\}$

(i) Draw M_4.

(ii) Find a determinstic finite automaton (FA) which accepts the same language as M_4.

[10 marks]

(b) Let: $\Sigma = \{a, b\}$
$L_5 = \text{Language}(\mathbf{a}(\mathbf{a} + \mathbf{b})^*)$
$L_6 = \text{Language}((\mathbf{ba})^*)$
$L_7 = \text{Language}(\mathbf{a}^* + \mathbf{b}^*)$

Give a string which:

(i) is in L_5 but not in L_6 or L_7

(ii) is in L_6 but not in L_5 or L_7

(iii) is in L_7 but not in L_5 or L_6 [6 marks]

Question 6. 24 marks

(a) State Kleene's Theorem. [6 marks]

(b) Explain how to construct a recursive scanner that recognizes the language accepted by a given regular grammar. [6 marks]

(c) Explain how to construct a regular expression that describes the language accepted by a given regular grammar. [6 marks]

(d) Describe a language that is *not* regular, and explain *informally* why it cannot be regular. [6 marks]

Question 7. 20 marks

(a) Define a pushdown automaton that accepts each of the following languages:

(i) The language $\{a^n b^m c^{m+n} \mid m, n > 0\}$.

(ii) The language consisting of strings over the alphabet $\{a, b\}$ with the same number of as and bs, but occurring in any order. [8 marks]

(b) Consider the CFG $G = (N, \Sigma, S, P)$ with productions:

$$\begin{aligned} S &\to STc \mid T \\ T &\to aT \mid U \\ U &\to aS \mid \Lambda \end{aligned}$$

(i) Define a pushdown automaton that accepts the language generated by G, using a **top-down** (expand-match) strategy.

(ii) Define a pushdown automaton that accepts the language generated by G, using a **bottom-up** (shift-reduce) strategy. [12 marks]

Question 8. 16 marks

(a) A Turing machine may be defined by a 6-tuple $(Q, \Sigma, \Gamma, \delta, q_0, F)$.

Explain each of the following:

(i) Σ

(ii) Γ

(iii) δ [6 marks]

(b) Let T be a Turing machine with input alphabet Σ.

(i) Define $accept(T)$.

(ii) Define $reject(T)$.

(iii) Define $loop(T)$.

(iv) Define "computably enumerable language". [10 marks]

Question 9.   12 marks

The following is (part of) a definition of a semantic function $\mathcal{V}$ for evaluating Boolean expressions:

$$\mathcal{V}(p \textbf{ and } q) = \begin{cases} \mathcal{V}(q), & \text{if } \mathcal{V}(p) = 1 \\ 0, & \text{otherwise} \end{cases}$$

$$\mathcal{V}(\textbf{not } p) = \begin{cases} 1, & \text{if } \mathcal{V}(p) = 0 \\ 0, & \text{otherwise} \end{cases}$$

$$\mathcal{V}(\textbf{true}) = 1$$

$$\mathcal{V}(\textbf{false}) = 0$$

Extend the definition to handle **Boolean variables**, v, whose values are determined by a store. [12 marks]

****