Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

The Implementation and Power of Computer Languages

Monday 26th January 2009

Time: 14:00 – 16:00

Please answer THREE questions from the FOUR provided:-

Question ONE and question THREE are compulsory

Use separate answerbooks for EACH section

The use of electronic calculators is NOT permitted.
1. **COMPULSORY**

I want to define a format for expressions. They consist of names, numbers, brackets and operators. Names consist of any sequence of upper-case and/or lower-case letters. Numbers consist of any sequence of digits in the range 0 to 7. Brackets are ( and ), and have the usual meaning. The operators are '&', which is left-associative, and '! ', which is right-associative, and of lower precedence than '& '. Each expression must be terminated by a '.' and as usual, blanks (i.e. spaces, tabs or newlines) can occur before or after any component.

For example:

Fred & 62 & (b ! 0 ! cd) .

a ! b ! 54 & XX & yy .

a) (i) What are the parse trees for the example expressions above? (4 marks)

(ii) Give an example to explain what it means to say that an operator is non-associative. (1 mark)

b) Write Lex and Yacc code to recognise the form of expressions as described above. You should not try to evaluate or translate the expression, so your yacc will not need any actions, and your lex code will only need "return" actions. You do not need to give code for common routines such as yywrap, yyerror and main. (15 marks)
2.  

a) It is usually easier to write Regular Expressions to recognise a (simple enough) pattern than it is to write the equivalent BNF. Give examples to show the main features of Regular Expressions that make this true.

There are patterns that can be described by BNF but not by Regular Expressions. Give an example to show that this is true, and briefly explain why your example is correct. (5 marks)

b) Here is a fragment of a data file, containing information about exam results for students:

Bloggs-Whosit, Fred [ COMP10011=56/80, COMP10021=47.5, COMP10041=47.3%, COMP10051=62.5, MATH10111=46/100]
Smith, Alice [ PSYC10101=73.1, COMP10011=65.0%, COMP10031="car crash", COMP10041="medical", COMP10051="medical", COMP10061=41% ]
McJones, Bob Richard Henry [ COMP10061=10, COMP10021=, COMP10041="absent", COMP10011=, HIST11000= ]
Patel, Chris [ COMP10021=65, COMP10011=54.7/80, COMP10041=73%, COMP10051=42.0%, MATH10121=56.2% ]

Each student can take any number of exams, listed between [ ] characters. The "quotes" can be used to contain any text string. An actual mark can be a single number, in which case it is a percentage (and may be followed by a % character), or it can be followed by a / and the total it is out of. As usual, blanks (i.e. spaces, tabs or newline) can occur before or after any component, although the information for each student should start on a new line.

Create lex and yacc grammars that, together, will match a file containing data like the fragment above. You do not need to write anything other than (named) regular expressions and BNF patterns.

This description is informal, and so may be incomplete and ambiguous. If you notice any ambiguity in the description, state what it is, how you resolved it in your lex and yacc grammars, and why you resolved it that way. (15 marks)
Section B

3. **COMPULSORY**

a) i) Draw a DFA for the pattern (0|001|010). (2 marks)

ii) Write down three words over the alphabet \{0, 1, 2\} that are generated by the grammar

\[
S \rightarrow 0S2|A \\
A \rightarrow \varepsilon|1A2
\]

that involve all three digits 0, 1, 2. (2 marks)

iii) What is the difference between a PDA and an NPDA? Are NPDA more powerful than PDAs? (2 marks)

iv) Design a Turing machine with tape alphabet \{0, 1\} that accepts exactly those words in \{0, 1\} that have even length. (4 marks)

b) Let L be the language of words that are accepted by the following NFA with \(\tau\)-moves.

\[
\begin{array}{c}
1 \\
2 \\
3 \\
4
\end{array}
\]

\[
\begin{array}{c}
\tau \\
a \\
b \\
b
\end{array}
\]

i) Draw a DFA for L. (5 marks)

ii) Give a pattern for L. (5 marks)
4. a) Give a context-free grammar for the language

\[ L = \{0^{2^n}1^n \mid n \in \mathbb{N}\} \]  

(4 marks)

b) Explain why the language \( L \) of a) is not regular.  

(5 marks)

c) Consider the Turing machine given by the following transition table:

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>( \sqcup )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>(1, ( \sqcup ), R) stop</td>
<td>stop</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>(2, ( \sqcup ), R) (0, b, N) stop</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>(2, a, R) (2, b, R) (3, ( \sqcup ), L)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>(4, ( \sqcup ), L) stop</td>
<td>stop</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>stop</td>
<td>(5, ( \sqcup ), L) stop</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>(5, a, L) (5, b, L) (1, ( \sqcup ), R)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The start state is 0 and the only accepting state is 1.

Give the configurations the machine goes through when started in state 0 on the left most letter of aaababa.  

(3 marks)

d) Which of the following words are accepted by the Turing machine in (c)?

- aba
- aaba
- aaabbaa
- aaababa

(4 marks)

e) What is the language accepted by the Turing machine in (c)?

(4 marks)