School of Mathematical and Computing Sciences
COMP 202 : Formal Methods of Computer Science

Test (21 August, 1997)

Time allowed = 90 minutes
Total marks = 50
Answer ALL questions

1. (a) Write a Regular Expression to describe the language in which every string consists of either an even number of as followed by an odd number of bs or an odd number of as followed by an even number of cs. [4 marks]

   (b) Write a Regular Expression to describe the language consisting of all strings of as, bs and cs, where as occur in multiples of two, bs occur in multiples of three, and consecutive groups of as or bs are separated by one of more cs. For example, "aa", "$cbbb$", "$cccc$", "$aacbbbcccaaaac$" and "$aaccaacbbbcbbbbbb$" are in this language, but "a", "bb", "aca" and "$aaab$" are not. [4 marks]

   (c) Write an Extended Regular Expression to describe the language consisting of lists of one or more *items*, separated by commas, where each *item* is either a number (a non-empty sequence of digits), two numbers separated by a colon, or three numbers separated by colons. For example, "123", "1, 2, 3" and "1, 2 : 3, 23 : 1 : 52, 1" are in this language, but "1, , 2", "1 : 2," and "1 : 2 : 3 : 1" are not. You may write d to stand for any digit. [4 marks]

   An Extended Regular Expression is like an ordinary Regular Expression, except that it can also use the following notations:

   $[e]$ means "zero one or more occurrences of e" (i.e. $[e] = e|\lambda$)
   e^+ means "one or more occurrences of e" (i.e. $e^+ = ee^*$)
   e^n means "n occurrences of e" (i.e. $e^n = \underbrace{e \frown \cdots \frown e}_{n}$)
   e^n_m means "at least m and at most n occurrences of e" (i.e. $e^m \frown [e]^{n-m}$)

   Your answer should make best use of these extensions.

2. Consider the regular expression $E = a^*(b|c)^* \mid b^*(a|c)^*$.

   (a) Draw a transition diagram for the simplest NFA you can construct to recognise the language defined by E. [4 marks]

   (b) Give a trace showing the behaviour of your DFA with $aacabc$ as input.
   Make sure that at each step you show the states that can be reached by following null transitions. [2 marks]

(c) Write a left-regular grammar equivalent to E. [3 marks]

(d) Draw a transition diagram for the NFA obtained by applying **either** the "top down" **or** the "bottom up" construction (as described in the Course Notes) to E.

Indicate clearly which construction you are using. [4 marks]

(e) Draw a transition diagram for the DFA obtained by applying the subset construction to your NFA from part (d). Show the correspondence between states in your DFA and sets of states in the NFA. [4 marks]

(f) Identify any states in your DFA in part (e) that can be eliminated to obtain a smaller DFA. [1 marks]

3. Consider the language PAL of palindromes over $\{a, b, c\}$, i.e. the set of all strings $\alpha \in \{a, b, c\}^*$ such that $\alpha^R = \alpha$.

   (a) Prove that PAL is not regular, using the Myhill-Nerode Theorem. [4 marks]

   (b) Write a Context Free Grammar that defines PAL. Give a brief explanation of why your grammar defines the required language. [4 marks]

4. Well-formed formulas of propositional logic (known as *wffs*) can be defined as follows:

   (i) The propositional constants, $true$ and $false$, are both wffs.

   (ii) Every propositional variable is a wff.

   (iii) If X and Y are wffs, then $\neg X$, $X \wedge Y$, $X \vee Y$, $X \Rightarrow Y$, $X \equiv Y$ and (X) are also wffs.

   (iv) Nothing else is a wff.

   In order to avoid ambiguity, we adopt the following conventions for the precedence and associativity of the propositional connectives:

Connective(s)	Precedence	Associativity
$\Rightarrow$, $\equiv$	1	Non-associative
$\vee$	2	Left-associative
$\wedge$	3	Left-associative
$\neg$	4	Associative

   As usual, parentheses are treated as having higher precedence that any of the propositional connectives.

   (a) Write a Context Free Grammar for wffs, which will give every wff a unique parse tree reflecting these conventions for precedence and associativity. [8 marks]

   (b) Draw a parse tree for each of the following wffs (where p, q and r are assumed to be propositional variables): [4 marks]

   (i) $p \wedge q \Rightarrow p \vee q$

   (ii) $(p \wedge q \Rightarrow r) \equiv (p \Rightarrow r) \wedge (q \Rightarrow r)$