Software Engineering 4I03 Michael Soltys

DAY CLASS
DURATION OF EXAMINATION: 2 Hours
MCMASTER UNIVERSITY FINAL EXAMINATION December 17, 2002

THIS EXAMINATION PAPER INCLUDES 2 PAGES AND 6 QUESTIONS. YOU ARE RESPONSIBLE FOR ENSURING THAT YOUR COPY OF THE PAPER IS COMPLETE. BRING ANY DISCREPANCY TO THE ATTENTION OF YOUR INVIGILATOR.

No aids allowed.

All questions worth 10 marks, for a total of 60.

1. A Coke machine offers the following choices, in can and bottle: Coke, Sprite, C-Plus, and Vanilla-Coke. It only takes loonies and quarters; a can is the equivalent of \$1, and a bottle is the equivalent of \$1.50. You must insert the proper amount before you make a selection (assume that you select first can or bottle, and then the beverage).

   Design a DFA that models the behavior of this machine (the alphabet would contain symbols for quarters, loonies, can, bottle, and the different beverages). An accepting string would contain the correct change first, then the selection can/bottle, and finally the beverage.

2. We have seen that PDAs accept exactly the same languages as CFGs. Explain why it is crucial for the PDAs to be ε-NFAs with a stack, and not just DFAs with a stack, in order to simulate arbitrary CFGs. That is, explain the role of nondeterminism in the simulation of CFGs by PDAs.

3. Consider only those C programs which take one input in binary, and have two possible outputs: accept and reject. Let $L = L(C)$ if on input $x \in \{0,1\}^*$, C accepts x if $x \in L$. Let $\mathcal{C}$ be the class of languages which can be accepted with such C programs. What is $\mathcal{C}$ in terms of classes of languages we have seen in the course? Justify your answer.

4. Design a Turing machine, which on input x (presented in binary), computes $3x + 5$ (presented in binary) and halts. Assume that the TM has an input tape containing x, and an output tape which, when the machine halts, will contain $3x+5$. Specify all the other tapes (if any), states, symbols, and give a table for the transition function.

5. Suppose that L is a recursive language. Show that L^* is also recursive.

Continued on Page 2

6. An *enumerator* is a Turing machine which takes no input, and when switched on, starts listing (enumerating) the strings in a language, by writing them on an output tape. It uses a special symbol, for example "$*$", to indicate where one string ends and another begins, so at any given point the output tape contains $x_1 * x_2 * x_3 * \ldots * x_i$. Note that some strings may repeat; that's OK.

   We say that an enumerator E lists the strings in L, denoted $L = L(E)$, if every string in L eventually appears on the list. That is, if $x \in L$, then there will come a moment when $x_i = x$ will be appended to the list. Note that x may appear on the list with repetion, but this does not matter as long as it appears at least once if it is in L.

   Show that a language L is recursive iff there exists an enumerator E that lists the strings of L in non-decreasing order of lengths (that is, if $\cdots * x_i * x_{i+1} * \cdots$ are consecutive strings on the list, then $|x_i| \leq |x_{i+1}|$).

The End