CS331 Autumn 2000 End-term Examination

Time: 3 hrs Total marks: 100

- ***The exam is open book and notes.***
- ***Results/proofs covered in class/problem sessions/assignments may simply be cited, unless specifically asked for.***
- ***Unnecessarily lengthy solutions will be penalized.***
- ***If you need to make any assumptions, state them clearly.***

1. *[15+5 marks]* A palindrome is a string x that reads the same forward and backward, i.e., $x = x^R$.
   (a) Given a context-free language L, show that it is undecidable whether L contains a palindrome.
   (b) Given a regular language L, is it decidable whether L contains a palindrome? Justify your answer with an undecidability proof or explain why the problem is decidable.

2. *[15 marks]* Given two CFLs L_1 and L_2 and a regular language R **that is neither** ϕ **nor** Σ^*, show that it is undecidable whether $L_1 \cap L_2 \subseteq R$. Assume Σ to be the alphabet for each of L_1, L_2 and R.

3. *[15 marks]* A string x is said to be *scrambled* in string y if:
   - $|y| \geq |x|$, and
   - there exists an injective function $f : \{1, 2, \ldots |x|\} \rightarrow \{1, 2, \ldots |y|\}$ such that the i^{th} symbol of x is the same as the $f(i)^{th}$ symbol of y. Informally, the symbols of x are scrambled and then thrown in between other symbols from the alphabet to form string y.

   For example, if $\Sigma = \{a, b, c\}$, string *abbc* is scrambled in *caabccba*. If we index symbols in each string from left to right, a possible function f for this example is: $f(1) = 8$, $f(2) = 4$, $f(3) = 7$ and $f(4) = 5$.

   Given alphabet Σ and a Gödel numbering scheme gn, define the predicate $\mathsf{Scrambled}(m, n)$ on natural numbers to be true if and only if $gn^{-1}(m)$ is scrambled in $gn^{-1}(n)$. Show that $\mathsf{Scrambled}(m, n)$ is primitive recursive.

   You must show primitive recursiveness of all predicates and functions you use other than the basic comparison predicates (i.e. $=, <, >, \leq, \geq$*), and the following functions:* $length(n)$, $power(m, n) = m^n$, $product(m, n) = m.n$, $sum(m, n) = m + n$, $monus(m, n) = m \dot{-} n$, $sign(n)$ *and* $prime(n) = n^{th}$ *prime.*

4. *[5 + 10 marks]* Consider the function $f(n) = \lfloor \sqrt(n) \rfloor$ for $n \in \{0, 1, \ldots\}$.
   (a) Assuming $f(0) = 0$, show that $f(n)$ is primitive recursive.
   (b) Using the flow-chart like notation for composing TMs discussed in class, design a (possibly multi-tape) TM that does the following:
   Given the string 01^n0 on tape 1, the TM halts with string $01^{f(n)}0$ on tape 2.

5. *[5 + 5 marks]* Consider languages (problems) $L_1, L_2, L_3 \subseteq \Sigma^*$, such that L_1 can be reduced to L_2 and L_2 can be reduced to L_3.

(a) Suppose it is further known that L_2 is recursive. Which of L_1 and L_3 must necessarily be recursive? Justify your answer briefly.

(b) Suppose we only know that L_2 is recursively enumerable. Which of L_1 and L_3 must necessarily be recursively enumerable? Justify your answer briefly.

6. *[15 marks]* Consider the PDA and DFA in Fig. 1. The PDA is assumed to start operation with X as the only symbol in the stack. Let L_1 be the language accepted by the PDA by final state and let L_2 be the language accepted by the DFA. The alphabet for both languages is $\Sigma = \{0, 1\}$. Define $L_1/L_2 = \{w \in \Sigma^* \mid \exists u \in L_2 \text{ such that } wu \in L_1\}$. Construct a PDA for L_1/L_2 or prove otherwise that L_1/L_2 is context-free. Outline a generalization of your method for an arbitrary PDA and DFA.

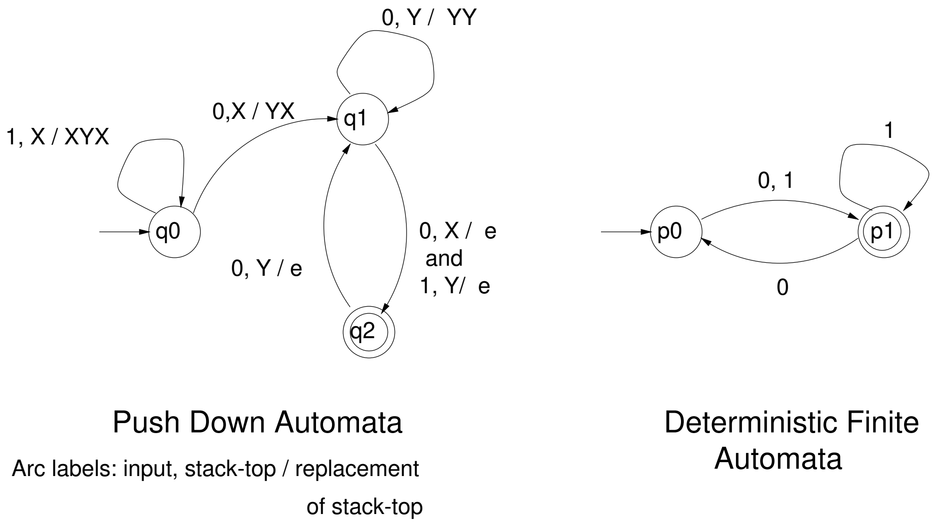

Figure 1:

7. *[10 marks]* For any language L, define the *Kleene-star* L^* as $\{x_1 x_2 \ldots x_k \mid k \geq 0 \text{ and } x_1, x_2 \ldots x_k \in L\}$. We know that **NP** is the class of languages for which if $x \in L$, membership can be determined in time polynomial in the size of the input by non-deterministic Turing Machines (NTM). Is the class **NP** closed under Kleene-star, i.e., if L is in **NP**, is L^* also necessarily in **NP**? Justify your answer.