Weighted Finite-State Transducers in Speech and Natural Language Processing
Problem Set 6:
Practice Final Examination

July 23, 2009

Directions: Explain your answers and show your work. Submit all computer code (e.g., Python scripts) used in completing the problems. Your completed exam must be submitted at the start of the Übung on Thursday, July 30.

1. For the example on the board, give three shortest path trees other than that shown.

2. Let \( G = (V, E) \) be a weighted, directed graph that contains no negative-weight cycles. Let \( s \in V \) be the source vertex, and let \( G \) be initialized by the Initialize-Single-Source(\( G, s \)) procedure defined in class. Prove that there exists a sequence of \( |V| - 1 \) relaxation steps that produces \( d[v] = \delta(s, v) \) for all \( v \in V \).

3. Arbitrage is the use of discrepancies in currency exchange rates to transform one unit of currency into more than one unit of the same currency. For example, suppose that 1 U.S. dollar buys 46.4 Indian rupees, 1 Indian rupee buys 2.5 Japanese yen, and 1 Japanese yen buys 0.0091 U.S. dollars. Then by converting currencies, a trader can start with 1 U.S. dollar and buy \( 46.4 \times 2.5 \times 0.0091 = 1.0556 \) U.S. dollars, thus turning a profit of 5.56%.

Suppose that we are given \( n \) currencies \( c_1, c_2, \ldots, c_n \) and a \( n \times n \) table \( R \) of exchange rates, such that one unit of currency \( c_i \) buys \( R[i, j] \) units of currency \( c_j \).

(a) Give an efficient algorithm for determining whether or not there exists a sequence of currencies \( < c_i, c_{i_2}, \ldots, c_{i_k} > \) such that

\[
R[i_1, i_2] \cdot R[i_2, i_3] \cdots R[i_k-1, i_k] \cdot R[i_k, i_1] > 1.
\]

Analyze the running time of your algorithm.
(b) Give an efficient algorithm to print out such a sequence if one exists. Analyze the running time of your algorithm.

4. Which of the following are regular languages? Prove your answer:
   (a) \{0^{2^n} : n \geq 1\}.
   (b) \{0^n : n is a primary number\}.
   (c) The set of all strings without three consecutive zeros.
   (d) The set of all strings with an equal number of 0s and 1s.

5. Construct a grammar for the language \( N(M) \) where
   \[ M = (\{q_0, q_1\}, \{0, 1\}, \{Z_0, X\}, \delta, q_0, Z_0, \emptyset), \]
   and \( \delta \) is given by
   \[
   \begin{align*}
   \delta(q_0, 1, Z_0) &= \{(q_0, XZ_0)\}, & \delta(q_0, \epsilon, Z_0) &= \{(q_0, \epsilon)\}, \\
   \delta(q_0, 1, X) &= \{(q_0, XX)\}, & \delta(q_1, 1, X) &= \{(q_1, \epsilon)\}, \\
   \delta(q_0, 0, X) &= \{(q_1, X)\}, & \delta(q_1, 0, Z_0) &= \{(q_0, Z_0)\}.
   \end{align*}
   \]

6. Let \( \Sigma \) be the set of lower case letters from the English alphabet. Implement a pushdown automaton that accepts strings from the language \( L = \{wcw^R : w \in (\Sigma - \{c\})^*\} \), where \( w^R \) denotes the string \( w \) in reverse order.

7. Based on the given Python scripts, complete the methods `getStartProduction` and `getProductionList` in the class `ContextFreeGrammar`.

8. Based on the given Python scripts, complete the method `isEqual` in the class `ContextFreeProductionList`.

9. Based on the given Python scripts, complete the `while` loop in the method `FSACharacteristicMachine::construct`. 