CS 39
Fall 2006
Theory of Computation

Homework 1
Due Oct 4, 2006

Prof. Amit Chakrabarti
Computer Science Department
Dartmouth College

Please think carefully about how you are going to organise your answers *before* you begin writing. Make sure your answers are complete, clean, concise and rigorous.

1. **(Operations on languages)** For the problems below, if you are specifying a set (i.e., a language), it should be specified in proper set notation as a single line answer in as simple a form as possible.

   1.1. For a language L over alphabet Σ, define $\textsc{Cycle}(L) = \{x_1x_2 : \exists\, x_1, x_2 \in \Sigma^* \text{ such that } x_2x_1 \in L\}$. If $A = \{0^n1^n : n > 0\}$, what is $\textsc{Cycle}(A)$? [7 points]

   1.2. We say that a string u is a *proper prefix* of string v if u is a prefix of v and $u \neq v$. For a language L over alphabet Σ, define

   $$\begin{aligned} \textsc{Min}(L) &= \{x \in \Sigma^* : x \in L \text{ and no proper prefix of } x \text{ is in } L\}\,. \\ \textsc{Max}(L) &= \{x \in \Sigma^* : x \in L \text{ and } x \text{ is not a proper prefix of any string in } L\}\,. \end{aligned}$$

   Specify an *infinite* language L such that $\textsc{Min}(L) = \textsc{Max}(L) = L$. [7 points]

   1.3. Is there an alphabet Σ and a language L over Σ such that $\textsc{Min}(L) = \Sigma^*$? Justify your answer. (No credit for an answer without correct justification.) [7 points]

   1.4. For a language L over alphabet Σ, define $\textsc{Half}(L) = \{x \in \Sigma^* : \exists\, y \in \Sigma^* \,(|x| = |y| \text{ and } xy \in L)\}$. For $L = \{0^p1^q0^r : q = p + r\}$, what is $\textsc{Half}(L)$? [7 points]

   1.5. For a language L over alphabet Σ, define $\textsc{HalfPalindrome}(L) = \{x \in \Sigma^* : xx^{\mathcal{R}} \in L\}$. For $L = \{0^p1^q0^r : q = p + r\}$, what is $\textsc{HalfPalindrome}(L)$? [7 points]

2. **(Tables $\leftrightarrow$ Diagrams)** Do Problem 1.2 and Problem 1.3 (Page 83) from Sipser's book. [10+5 points]

3. **(Designing DFAs)** For each of the languages below, over the alphabet $\{0, 1\}$, specify a DFA. Use a state diagram to specify your DFAs unless either the number of states is too large or the diagram too clumsy. In such cases, specify specify the DFA formally, using appropriate mathematical notation.

   Keep your DFAs simple; unnecessarily complicated DFAs will lose credit.

   3.1. The language that contains only the empty string. [3 points]

   3.2. The set of all strings in $\{0, 1\}^*$ except 11 and 111. [5 points]

   3.3. $\{x \in \{0, 1\}^* : x \text{ does not contain the substring } 110\}$. [7 points]

   3.4. The set of all strings in $\{0, 1\}^*$ such that each string is of length at least three and every block of three consecutive symbols has at least one 0. (Thus, for example, 0011001 is in the language, but 0011100 is not.) [10 points]

3.5. The set of all strings such that each string, when interpreted as a binary number, is divisible by 5. (For example, 11001 is in this set because, when interpreted as a binary number, its value is 25 which is divisible by 5.) [10 points]
Hint: For integers m, n, p we have $(mn + p) \bmod 5 = ((m \bmod 5) \cdot n + p) \bmod 5$.

3.6. The set of all strings such that each string is of length at least 100 and every block of 100 consecutive symbols has at least ten 0's. [15 points]
Hint: It may help to redo Problem 3.4, specifying your DFA formally.