8 Homework

Due: Thursday, March 21, 2013.

Instructions

- Please, review the homework grading policy outlined in the course information page.

- On the first page of your solution write-up you must make explicit which problems are to be graded for regular credit, which problems are to be graded for extra credit, and which problems you did not attempt. Use a table that looks like this:

<table>
<thead>
<tr>
<th>Problem</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit</td>
<td>RC</td>
<td>RC</td>
<td>RC</td>
<td>EC</td>
<td>RC</td>
<td>EC</td>
<td>NA</td>
<td>NA</td>
<td>EC</td>
<td>...</td>
</tr>
</tbody>
</table>

where “RC” denotes “regular credit”, “EC” denotes “extra credit”, and “NA” denotes “not attempted”. Failure to include such a table will result in an arbitrary set of problems being graded for regular credit, no problems being graded for extra credit, and a 5% penalty assessment.

- You must also write down with whom you worked on the assignment. If this varies from problem to problem, write down this information separately with each problem.

Problems

Required: 5 of the following 7 problems
Points: 20 points per problem

1. Do the Problem 4.2

Consider the problem of determining whether a DFA and a regular expression are equivalent. Express this problem as a language and show that it is decidable.
2. (a) Do the Problem 4.4
Let \( A_{\text{CFG}} = \{ < G > \mid G \text{ is a CFG that generates } \epsilon \} \). Show that \( A_{\text{CFG}} \) is decidable.

(b) Do the problem 4.7
Let \( B \) be the set of all infinite sequences over \( \{0, 1\} \). Show that \( B \) is uncountable using a proof by diagonalization.

3. Do the Problem 4.11
Let \( \text{INFINITE}_{\text{PDA}} = \{ < A > \mid A \text{ is a PDA and } L(A) \text{ is an infinite language} \} \). Show that \( \text{INFINITE}_{\text{PDA}} \) is decidable.

4. Do the Problem 4.13
Let \( A = \{ < R, S > \mid R \text{ and } S \text{ are regular expressions and } L(R) \subseteq L(S) \} \). Show that \( A \) is decidable.

5. Do the Problem 4.17
Prove that \( \text{EQ}_{\text{DFA}} \) is decidable by testing the two DFAs on all strings up to a certain size. Calculate a size that works.

6. Do the Problem 4.21
Let \( S = \{ < M > \mid M \text{ is a DFA that accepts } w^k \text{ whenever it accepts } w \} \). Show that \( S \) is decidable.

7. Do the Problem 4.30
Let \( A \) be a Turing-recognizable language consisting of descriptions of Turing machines \( \{ < M_1 >, < M_2 >, \ldots \} \), where every \( M_i \) is a decider. Prove that some decidable language \( D \) is not decided by any decider \( M_j \) whose description appears in \( A \). (Hint: You may find it helpful to consider an enumerator for \( A \).)