Final Exam — CSE 105, Spring ’03

Name (print): ____________________________
Student I.D.: ________________

• **IMPORTANT**: Before starting to work on any of the problems, print your name at the top of every page.

• This exam is closed book. You are only allowed to use one page of notes (double sided is fine)

• **No form of collaboration is allowed.** You are not allowed to look at other students solutions or note sheets, or ask questions to anybody other than the instructor and TAs. **Any violation will be severely punished.**

• Your solution will be evaluated both for correctness and clarity. A poorly written solution won’t get full credit even if correct.

• Read all the problems first before start working on any of them, so you can manage your time wisely

• **DO NOT PANIC!**

<table>
<thead>
<tr>
<th>Problem</th>
<th>Points</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td>60</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>+10</td>
<td></td>
</tr>
</tbody>
</table>

Problem 7 is for extra credit. The first 6 problems of this exam account for 40% of your final grade. So, if you get $X$ points out of 60, you will receive $\frac{2}{3}X$ credit toward your course grade. The points in the extra problem contribute in the same proportion, i.e., if you get $Y$ points in the problem set you will receive $\frac{2}{3}Y$ extra credit, contributing up to 6.666% additional credit to your course grade.
Problem 1 [10 pts]. Name: __________________________

Give a context free grammar for the following language:

\[ L = \{a^ib^ja^k : i, j, k \geq 0 \text{ and } k = i \cdot (j \mod 3) \} \]

(Remember, \( j \mod 3 \) is the remainder of dividing \( j \) by 3, e.g., 6 mod 3 = 0, 4 mod 3 = 1, 11 mod 3 = 2, 0 mod 3 = 0.) In your solution, for each non-terminal symbol, add a brief (say, one line) explanation of what is the language generated by that symbol.
Problem 2 [10 pts]. Name: __________________________

For each of the following languages, says if it is regular, context free (but not regular), decidable (but not context free), recognizable (but not decidable), co-recognizable (but not decidable), or neither recognizable nor co-recognizable. No proof is required. Just give your answer or best guess to each question. Each correct answer is 2 points. Each incorrect or missing answer 0 points.

\[
\begin{align*}
A &= \{a^i b^j c^{i+j} : i, j \geq 5\} \\
B &= \{\langle M \rangle : M \text{ is a DFA and } M \text{ accepts } \langle M \rangle \} \\
C &= \{a^i b^i c^k : i = j = (k \mod 3)\} \\
D &= \{\langle R, G \rangle : \langle R \rangle \in ALL_{RE} \text{ or } \langle G \rangle \in ALL_{CFG} \} \\
E &= \{\langle M, x, y \rangle : M \text{ is a TM that accepts } x, \text{ but does not accept } y \}
\end{align*}
\]

<table>
<thead>
<tr>
<th>Language</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regular</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Context Free, but not Reg.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Decidable, but not C.F.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Recog., but not Dec.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Co-recognizable, but not Dec.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neither recog. nor corecog.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Problem 3 [10 pts]. Name: ___________________________

Design a DFA with (at most) 4 states accepting the language

\[ L = \{ w \in \{a,b\}^* : \text{every } a \text{ in } w \text{ is followed by an odd number of } b's \} \]

For example, \( abbbabab, bbab, bab \in L \), but \( a, ababbbabb, bbbababb \notin L \). In order to receive full credit, you should give a solution with no more than 4 states. If you can’t, give a correct solution with more than 4 states for partial credit.
Problem 4 [10 pts]. Name: ____________________________

Circle the correct answer for each of the following questions. Correct answer gives 1 point, wrong answer 0 points.

T F If $A \leq B$ then $\overline{A} \leq B$

T F If $A \leq_m B$ then $\overline{A} \leq_m B$

T F If $A \leq_m B$ then $\overline{B} \leq \overline{A}$

T F If $A \leq B$ and $B \leq_m \overline{C}$, then $A \leq C$

T F If $A \leq B$, then $A \leq_m B$.

T F $A_{DFA} \leq_m E_{CFG}$

T F $EQ_{CFG} \leq E_{PDA}$

T F $EQ_{TM} \leq_m E_{TM}$

T F $EQ_{DFA} \leq A_{CFG}$

T F $A_{CFG} \leq EQ_{DFA}$
Problem 5 [10 pts]. Name: _____________________________

Prove that $\overline{E_{TM}} \leq_m ALL_{TM}$, giving a map reduction from $\overline{E_{TM}}$ to $ALL_{TM}$.
Problem 6 [10 pts]. Name: _______________________

Transform the following NFA into an equivalent DFA completing the picture below using the procedure studied in class. (The states of the DFA are labeled with sets of states of the original NFA.)
Problem 7 [Extra credit]. Name: ________________

Before attempting this part, make sure you have satisfactorily solved all the other problems. This problem will be graded only if your score in the rest of the exam is higher than 50 points (out of 60 total).

In class we studied various kinds of transformations between automata, e.g., from NFA to DFA, and observed that they are all algorithmically computable. This is not always the case. Consider the problem of transforming a PDA into a DFA. Of course, not all PDAs can be transformed into DFAs, because some of them define non-regular languages. So your transformation, given a PDA $P$, should output a DFA $D$ such that $L(D) = L(P)$ if $L(P)$ is regular, or a special error message (“Non regular language!”) if $L(D)$ is not regular. Prove that the above problem cannot be solved algorithmically, i.e., no computer program can perform that transformation. (Notice, the problem description above is a bit fuzzy. In particular, we didn’t formulate the problem as a language, and when $L(D)$ is regular, we didn’t specify which of the many equivalent DFAs should be output. This is intentional. This is the kind of descriptions of problems you might encounter in real life: given the description of a stack-based computer device, figure out if the stack is really necessary, and if not, give an alternative implementation that does not use the stack.)