Name: ____________________

Student ID: ____________________

CS181 Winter 2011 - Final Exam

Due Friday, March 11, 2011, at 4:00pm, in box A11 in BH2432

- This exam is open-book and open-notes, but any materials not used in this course are prohibited, including any material found on the internet. Collaboration is, of course, also prohibited. **Please avoid temptation by not working on the examination while you are in the presence of any other student who has taken or is currently taking CS 181.** You are allowed to use any theorem shown in class or in the textbook, as long as you clearly cite it.
- You are limited to 10 hours (not necessarily contiguous) to take this exam. It must be turned in by 4:00 P.M. on Friday, March 11, 2011. Please submit to box A11 in room 2432BH.
- Place your name and UID on every page of your solutions. **Please use separate pages for each question. All problems require rigorous proofs.**
- For each part (except for the extra credit), 20% of the points will be given if instead of an answer, your write "I don't know".

Honor Code Agreement: I understand this exam is open-book and open-notes, but any materials not used in this course are strictly prohibited. I also understand that this exam is to be taken individually without any outside help (except possibly from the professor or the TA) within the time limits set forth. I agree to adhere to the course honor code and if I am unsure of any rules of the honor code, I will ask for clarification from the professor or the TA.

Signature: ____________________

Question	**Points**	
1		60
2		50
3		30
4		35
EC		40
Total		

Name: ____

Student ID: ____

1. We say that a TM is n-bounded (and write n-TM) if the machine has at most n states and its tape alphabet has at most n symbols, that is: $|Q| \leq n$ and $|\Gamma| \leq n$.

   (a) (**15 pts.**) Prove that there exists a language $L \in R$ such that no 139-TM accepts L.

   For each one of the following, determine whether
   $L \in R$, $L \in RE \setminus R$ or $L \notin RE$:

   (b) (**15 pts.**)
   $L_1 = \{\langle M\rangle \mid M$ is a Turing Machine and there exists a 139-TM M' such that $L(M) = L(M')\}$

   (c) (**10 pts.**)
   $L_2 = \{\langle M\rangle \mid M$ is a Turing Machine and there exists a 139-TM M' such that $L(M) \neq L(M')\}$

   (d) (**20 pts.**) $L_3 = \{(\langle N\rangle, x) \mid N$ is a 139-TM, and N accepts $x\}$
   (Hint: Minsky discovered a universal turing machine that has only 7 states and 4 symbols in its tape alphabet.)

Name: ____________________________________

Student ID: ____________________________________

2. Let a *length-selected Turing Machine (lsTM)* $\mathcal{M}$ be a countably **infinite** sequence of Turing Machines $(M_0, M_1, M_2, \ldots)$. On input x, the way that $\mathcal{M}(x)$ works is as follows: First, it looks at the length of the input x. If the length of x is k, then it executes $M_k(x)$, and outputs whatever M_k outputs. For instance, on input 101, the $lsTM$ $\mathcal{M} = (M_0, M_1, M_2, \ldots)$ would execute $M_3(101)$.

   (a) (**15 pts.**) Show that there are an uncountable number of $lsTM$s.
   (Hint: Recall that when we used diagonalization to prove that the real numbers $\mathbb{R}$ are uncountable, we used the (countably infinite) decimal places of real numbers to do the diagonalization. What should take the place of decimal places for lsTMs?)

   (b) (**15 pts.**) Are there any languages that are not decidable by $lsTM$s? Either construct a language that is not decidable by any $lsTM$, or prove that every language is decidable by some $lsTM$.

   Now we define *k-$lsTM$* $\mathcal{M}$ as the *finite* sequence of Turing Machines $(M_0, M_1, \ldots, M_{k-1})$. On input x, $\mathcal{M}$ checks the length of the input, calculates $\ell = (|x| \mod k)$ and then executes $M_\ell(x)$.

   We say that a machine $\mathcal{M}$ is a *modular-$lsTM$* if there exists some $k \geq 0$ such that $\mathcal{M}$ is a *k-$lsTM$* machine.

   (c) (**20 pts.**) Prove that *modular-$lsTM$* machines are not as powerful as $lsTM$ machines. In other words, show there exists a language L which no *modular-$lsTM$* can decide, but there exists an $lsTM$ which decides L.

Name: ______________________________

Student ID: ______________________________

3. Let a *cyclic consuming DFA (ccDFA)* be a DFA with the following changes:

   (a) While processing the input, the ccDFA can choose whether or not it "consumes" each input character. If the ccDFA consumes the input character, it is lost forever, just like for ordinary DFAs. When the ccDFA has finished processing the input, a special character "$\circlearrowright$" appears, and then whatever input characters haven't been consumed appear *again*, for the ccDFA to process again. This process continues indefinitely.

   (b) A ccDFA has a set C of consuming states – if the ccDFA transitions to a state in C, then the character just processed is consumed. A ccDFA also has a set F of Halting Accepting States, along with a disjoint set R of Halting Rejecting States.

   (c) The transition function δ of a ccDFA maps $(\Sigma \cup \{\circlearrowright\}) \times Q$ to Q.

   Example: Consider a ccDFA M that first consumes all 0s that it sees and keeps track of whether it has consumed an even or an odd number of 0s, until it sees $\circlearrowright$. After that point, M checks to see if the remaining input is exactly 123. The machine accepts if it consumed an odd number of 0s and the remaining input was exactly 123, and otherwise rejects. On input 010203, the ccDFA M would see the following string of characters before accepting: "010203 $\circlearrowright$ 123".

   On any given (finite) input string x, a cyclic DFA can either accept it, reject it, or loop. We say that a cyclic DFA decides a language L if it always halts and accepts exactly those strings in L.

   (a) (**15 pts.**) Give an example of a language L that is context free but **not** regular, and is decidable by a ccDFA.

   (b) (**15 pts.**) Give an example of a language L' that is **not** context free, but is decidable by a ccDFA.

Name: ______________________________

Student ID: ______________________________

4. **(35 pts.)** Use the recursion theorem to show that

$$L = \{\langle M\rangle \mid M \text{ on input } \varepsilon \text{ outputs } \langle M\rangle\}$$

is not decidable.

Name: ______________________________

Student ID: ______________________________

5. **Extra Credit** (**40 pts.**) Let L be a regular language, and let B be some arbitrary language (which could even be undecidable!). Define $L \hookrightarrow B = \{x \mid \exists y \in B \text{ such that } xy \in L\}$.

   Prove that $L \hookrightarrow B$ is always regular.

   (Hint: Note you cannot hope to construct a machine for $L \hookrightarrow B$ explicitly, since we don't know of any kind of machine for B. Therefore you need to only prove that there must *exist such a machine.)*
   (Hint: Consider strings x, y_1, y_2 such that both xy_1 and xy_2 are in L, and yet $y_1 \in B$ but $y_2 \notin B$. According to the definition, is it the case that $x \in (L \hookrightarrow B)$? Think about what this means.)