HOMEWORK #8, CPSC 421/501, FALL 2017

JOEL FRIEDMAN

Please note:

(1) **We may only mark a subset of the problems below, depending on time constraints; the solution set we provide will solve all of the problems below.**
(2) Proofs should be written out formally. **Your solutions should be explained: e.g., if we ask for a DFA, you should explain how it works, not merely produce a diagram of the DFA.**
(3) Homework that is difficult to read may not be graded.
(4) You may work together on homework, **you must write up your own solutions individually.** You must acknowledge with whom you worked (specify their `ugrad.cs.ubc.ca` email addresses). You must also acknowledge any sources you have used beyond the textbook and two articles on the class website.
(5) When you submit your homework to `gradescope.com`, you need to put the solutions to different problems on different pages; `gradescope.com` will ask you to identify which pages correspond to which problems. Please use the problem numbers below.
(6) Bonus questions count for marks above the 10% homework grade.

Homework Problems

(1) Consider a fixed language, $L \subset \Sigma^*$, over an alphabet Σ, and consider all oracle Turing machines with oracle L (see class notes on November 8 or Definition 6.18 of [Sip]).
 (a) In Chapter 4 of [Sip] and in class we argued that the set of languages that Turing machines accept is countably infinite. Does the same argument work for "Turing machines" replaced with "Turing machines with oracle L" ? Explain.
 (b) Give an L such that the set of langauges accepted by Turing machines with oracle L is equal to set of langauges accepted by Turing machines. [Justify this claim about your choice of L.]

Research supported in part by an NSERC grant.

   (c) Give an L such that the set of langauges accepted by Turing machines with oracle L is not equal to set of langauges accepted by Turing machines. [Justify this claim about your choice of L.]
(2) Exercise 6.4 of [Sip].
(3) Prove the assertion regarding replacing $a_1 \vee \cdots \vee a_\ell$ by a 3CNF with $\ell - 2$ clauses at the top of page 311 of [Sip].
(4) Consider a non-deterministic Turing machine, M, that for each input w (1) M checks if w is the description of a Boolean formula, and (2) if so, so that $w = \langle f \rangle$ with $f = f(x_1, \ldots, x_n)$, M non-deterministically "guesses" true/false values of each of $x_1, \ldots, x_n$ and evaluates f. Say that M accepts w if $w = \langle f \rangle$ if $f(x_1, \ldots, x_n)$ evaluates to "true,", and that otherwise M rejects w (i.e., M rejects w if w is not of the form $\langle f \rangle$ or if $f(x_1, \ldots, x_n)$ evaluates to "false"). Say that M' works like M except that M' accepts w if $f(x_1, \ldots, x_n)$ evaluates to "false", and that otherwise M' rejects w.
   (a) Is (the description of) $f(x_1) = x_1$ accepted by M? Is it accepted by M'? Explain.
   (b) Is $f(x_1) = x_1 \wedge (\neg x_1)$ accepted by M? Is it accepted by M'? Explain.
   (c) Is $f(x_1) = x_1 \vee (\neg x_1)$ accepted by M? Is it accepted by M'? Explain.
   (d) Let UNSAT be the langauge of descriptions of Boolean formulas, f, that are not satisfiable, i.e., such that $f(x_1, \ldots, x_n)$ is false for all $(x_1, \ldots, x_n) \in \{T, F\}^n$. Does M or M' recognize UNSAT? Explain in terms of the above three examples of $f = f(x_1)$.

Department of Computer Science, University of British Columbia, Vancouver, BC V6T 1Z4, CANADA, and Department of Mathematics, University of British Columbia, Vancouver, BC V6T 1Z2, CANADA.

E-mail address: jf@cs.ubc.ca or jf@math.ubc.ca

URL: http://www.math.ubc.ca/~jf