HOMEWORK 5, CPSC 421/501, FALL 2019

JOEL FRIEDMAN

Please note:

(1) You must justify all answers; no credit is given for a correct answer without justification.
(2) Proofs should be written out formally.
(3) Homework that is difficult to read may not be graded.
(4) You may work together on homework, **but you must write up your own solutions individually.** You must acknowledge with whom you worked. You must also acknowledge any sources you have used beyond the textbook and two articles on the class website.

In these exercises, "the handout" refers to the article "Self-referencing, Uncountability, and Uncomputability" on the 421/501 homepage.

(1) Let $\Sigma = \{a, b\}$, and let L be the language over Σ

$$L = \{\sigma_1\sigma_2 \ldots \sigma_k \mid \sigma_1, \ldots, \sigma_k \in \Sigma,\ \sigma_1 = \sigma_k\};$$

i.e., L is the language of strings whose first symbol equals its last symbol.

(a) Describe a DFA recognizing L, and explain how it works.

(b) Based on your DFA, describe a Turing machine that decides L, and explain how it works. What is your work tape Γ, your accept state, and your reject state? [Your Turing machine should work mostly like the DFA, but there will be minor differences due to the way that Turing machines work; for example, Turing machines have an accept state and a reject state that indicates that the machine halts.]

Research supported in part by an NSERC grant.

(2) Let $\Sigma = \{0, 1, \#\}$, and let L be the language over Σ given as

$$L = \{s\#s \mid s \in \{0,1\}^*\};$$

for example, #, 10#10, and 000#000 are elements of L, but 10#0, ##, and 110#111 are not. Give a 1-tape Turing machine that decides L, and explain how it works.

(3) Let Σ, L be as in Problem 2. Give a 2-tape Turing machine that decides L in linear time, i.e., that if given an input that is a string of length n, halts in time $O(n)$ (i.e., for sufficiently large n, halts in time at most Cn for some constant C independent of n). Explain how your machine works.

(4) Let Σ, L be as in Problem 2. Use the Myhill-Nerode theorem to show that L is not regular.

(End of Homework Problems to be Submitted for Credit.)

Exercises Beyond the Homework (not for credit, solutions will not be released):

Turing Machines with a Countable Number of States

[Thanks to L.P. for this question.] Consider a Turing machine with a (countably) infinite number of tapes, instead of a finite number of tapes: you have a fininte number of states, Q, a given input alphabet, Σ, and a (finite) tape alphabet Γ, but for each $m \in \mathbb{N}$ you have "tape number m;" hence the transition function δ is a map

$$\delta\colon Q \times \Gamma^{\mathbb{N}} \to Q \times \Gamma^{\mathbb{N}} \times \{L, R, S\}^{\mathbb{N}}.$$

Show that any language, L, over Σ is decided by such a Turing machine.

Hints for Exercises Beyond: Hints appear on the page after this.

There may be many ways to do this; here is one: consider an algorithm that sweeps through the input, moving to the right, writing the first symbol of the input on tape 2, the second symbol of the alphabet on tape 3, etc., moving only the tape head of the input tape, and leaving all other tape heads on their first cell (can this be done with our conventions on infinite tape Turing machines?). When you finally reach a blank symbol on the input tape, you should be able to finish immediately (or almost immediately).

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

E-mail address: `jf@cs.ubc.ca`

URL: `http://www.cs.ubc.ca/~jf`