Lectures and Reading. Monday’s lecture will finish the proof of Kleene’s Theorem. For that, please read section 3 of Debray’s notes in-tandem with the GNFA coverage in Ogihara’s notes on Regular Expressions on the course webpage. Then for Wednesday, beginning coverage of the Myhill-Nerode Theorem, skim section 4 and focus on section 5 of Debray’s notes. Also read the Myhill-Nerode handout on the course webpage. You can read ahead to section 6 of Debray’s notes if you wish, but it will be “covered” only after Turing Machines are formally introduced in week 4.

(1) Suppose we change the rules of the “spears and dragons game” to read as follows:

- Each character in the input string—figuratively, each char is a “room” in a linear “dungeon”—is either $ for “spear,” $ for “dragon,” or $ for “lamp.”
- The Player $ may hold a maximum of 2 spears at any one time.
- Upon entering a room with a spear, $ may pick it up unless already carrying two spears.
- Upon entering a room with a dragon, if $ has no spear, $ is dead. Else, $ uses one spear to kill the dragon, and is then carrying one fewer spear.
- Upon entering a room with a lamp, if $ has killed two dragons since the last time $ picked up a spear (it follows that $ currently has no spear), then $ may rub the lamp and the genie in the lamp will give $ one spear. Otherwise, the lamp has no effect.

Design a deterministic finite automaton $ that simulates this game. In particular, $ should equal the language of strings representing “dungeons” that $ survives. The final state of $ should also tell the number of spears that $ has when exiting the dungeon. (If you sense an ambiguity in the rules, you may ask on Piazza, and/or you may describe the perceived ambiguity and detail the interpretation you took to resolve it. 18 pts.)

(2) Convert the following NFA $ with $-transitions into an equivalent DFA. The code for $ has $ = \{ 1, 2, 3, 4 \}$, $ = \{ a, b \}$,

$$
\delta = \{ (1, \epsilon, 2), (1, a, 3), (2, a, 2), (2, b, 4), (3, b, 2), (3, b, 4), (4, a, 4), (4, b, 1) \},
$$

$s = 1$, and $ = \{ 2 \}$. (18 pts.)

(3) Calculate a regular expression over $ = \{ a, b \}$ for the language of strings that are not accepted by the following NFA: $ = \{ s, q, f \}$, $ = \{ f \}$, and

$$
\delta = \{ (s, a, q), (s, b, f), (q, b, s), (q, a, f), (f, a, s), (f, \epsilon, q) \}.
$$

(Note that if the last instruction were on $ not $ it would be a DFA.) You must use a strategy based on theorems in lectures and posted notes, not just inspection (that is, “hacking”). (24 pts., for 60 total on the written part of the set)

(4) * There will be a further problem or sequence of problems for presentation or further discussion; I have not yet settled how to handle them; due times will vary.