Math 304 — Hour Exam 2

March 23, 2016

General Directions. This is an open-book, open-notes, open-computer test. However, you may not communicate with any person, except me, during the test. You have the full class period (50 minutes) in which to do the test. Put your answer to each question in the space provided (use the backs of pages if you need more space). Be sure to **show your work**! I give partial credit for incorrect answers if you show correct steps leading up to them; conversely, I do not give full credit even for correct answers if it is not clear that you understand where those answers come from. Good luck.

This test contains 3 questions (one with 2 parts) on 3 pages.

Question 1 (5 Points). Phineas Phoole observes that some context free grammars with only one variable generate regular languages. For example

$S \rightarrow aS \mid a$

only has one variable and generates the regular language aa*. From this and similar examples Phineas concludes that context free grammars must have at least 2 variables in order to generate non-regular languages. Show that Phineas is wrong, i.e., that there is some context free grammar with only one variable that generates a non-regular language.

Several classic examples of non-regular languages can be defined by context free grammars with only 1 variable, for example $a^n b^n$:

$S \rightarrow aSb \mid \epsilon$

Question 2. Consider the language L containing all and only strings of the form $a^n w w^R b^n$ where w is any string over {a,b}. (Recall that w^R denotes the reversal of string w.)

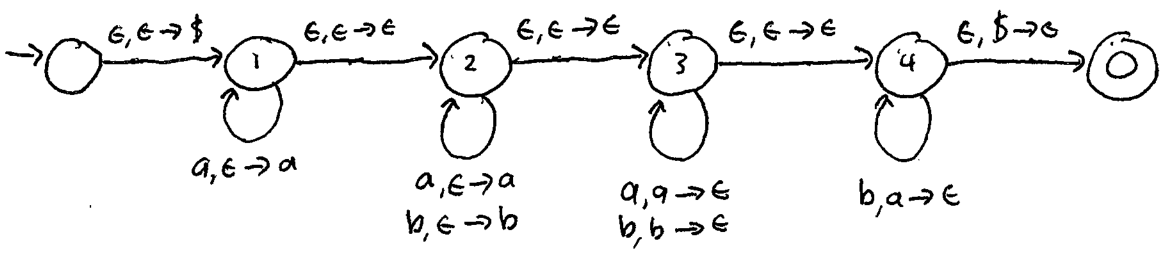
Part A (10 Points). Write a context free grammar that generates L.

This grammar combines an idiomatic set of rules for $a^n b^n$ with one for ww^R:

$$S \to aSb \mid R$$
$$R \to aRa \mid bRb \mid \epsilon$$

Part B (15 Points). Draw a pushdown automaton that recognizes L.

One quick way to answer this is to ~~us~~ use the book's construction of a PDA from a CFG on the answer to part A. Or, to design a custom PDA, consider one that works in 4 stages, non-deterministically guessing when to move between stages: (1) collect a on the stack, (2) collect a and b, (3) match a and b in input against ones on the stack, (4) match b in input against a on stack (stages 1 & 4 handle a^n and b^n; 2 & 3 handle w and w^R):



Question 3 (20 Points). Consider the set of strings of the form $xycx^Ry^R$ where x and y are any strings over {0,1}. For example, 10010c00101 is in this set because it has the required form if $x = 100$ and $y = 10$. Show that this set is not context free.

The following observation will be helpful later in this proof: if x and y are strings over {0,1}, then xy and x^Ry^R contain the same total number of symbols, the same number of 0s, and the same number of 1s.

Assume the set is context free. Then by the pumping lemma, every string in the set can be pumped into more strings still in the set. But consider the string $s = 0^p1^pc0^p1^p$, where p is the pumping length This string is in the set (with $x = 0^p$, $y = 1^p$), and is long enough to be pumped. There are several ways s can break into parts for pumping, namely...

Case 1: Both pumpable parts lie on the same side of the c. Then pumping produces a string in which the alleged xy part contains a different number of characters than the alleged x^Ry^R part, which by the observation above is impossible.

Case 2: One pumpable part contains the c. Then pumping up produces a string with too many cs to be in the set.

Case 3: One pumpable part lies before the c and the other after. By the pumping lemma requirement that $|vxy| \leq p$ (in this relation x and y refer to the substrings from the pumping lemma, all other uses of x and y in this answer refer to the substrings from the definition of the set), the pumpable part before the c contains only 1s and the part after the c contains only 0s. Pumping up then produces a string in which the alleged xy part contains more 0s than 1s, but the alleged x^Ry^R part contains more 1s than 0s, violating the observation above. The pumped string is thus not in the set.

These cases cover all ways of decomposing s for pumping, but none of them can yield strings in the set after pumping. This contradicts the assumption that the set is context free.