Discussion Section Notes on Pumping Lemma (PL)

Things to remember about using PL to prove a language L is not regular:

When choosing the word $w \in L$ to make the proof. Try to select a long word (one of the conditions of the PL says it must be at least p-characters long) that forces the automaton to remember arbitrarily long sequences or to count.

Here is an example:
If $L = \{ w \, : \, w = x||x,$ where x is a word in $\{ a, b \}^* \}$ that is, L is the language of all strings (or words) form by concatenating the same word twice. We want to prove L is not regular. Assume it is: M is the automaton that recognizes L.

Bad word to choose: $w = a^p a^p$
Why: This words does not make the lemma fail, since all partitions of w into xyz lead to words of the form $w' = xy^iz = a^s$ (for some even s) which can be seen as $a^{s/2}a^{s/2}$. So w' is in L. This choice doesn't work because an automaton M would not have to remember or count anything when reading this particular word w (though w is in L). In some sense, $w = a^p a^p$ is *not* one of the "hardest" words that make L hard to recognize for an automaton.

Good word to choose: $w = a^p b a^p b$
Why: How would an automaton check whether $w \in L$ or not? It would read (recognize) the first p characters (a^p), then b, and then would read the remaining characters to check if they are equal to what it read first. But, if the string is long enough M has no way to know when the first part ($x = a^p b$) ends! So, M must somehow count or "memorize" characters, which we know it can't do for a long time. Therefore, M must traverse a loop of states when recognizing w if $|w| \geq p$. That will give us enough help to come up with a new word w' that "fools" M and makes the lemma fail (one which does not use the loop or which uses it 2 or more times). In some sense, this new $w' \notin L$ will be accepted by M because M cannot tell the difference between w and w'.

Another way to see it, is that the automaton must loop when reading the first a^p part of w (remember that the pumping length p is the size of the automaton), So, if the automaton M is on a state (say q) which follows the loop in the execution path of w, M cannot precisely remember the number of $a's$ read because it can't remember how many loops were executed when reading w.

Ok. Let's see: first I assume L is regular, and I have p; then I choose a word longer than p. If I find a contradiction for a specific partition xyz of w, Am I done?

Unfortunately, no, you aren't. The lemma says that

"*for any word w long enough, THERE IS A partition of w such that ....*"

so, it is NOT enough to CHOOSE a *particular* partition x where the lemma fail, we must show that starting from ANY partition of the word (s.t. $|xy| \leq p$ and $|y| \geq 0$), we can create strings (ie. choose a $i \geq 0$) that fool M.

Example of wrong proof:
Let $L = \{ b^n a b^n \ : \ n \geq 0 \}$ be the language we want to prove nonregular. Assume it is, then by PL we have a pumping length p. Now we pick $w = b^{p-1} a b^{p-1}$. Clearly, $|w| \geq p$. In order to get the contradiction we say that when $x = b^{p-1}$, $y = a$ and $z = b^{p-1}$ (notice that $|xy| \leq p$, so it seems we are ok) we choose $i = 2$ to build the word $xy^2z = b^{p-1}aab^{p-1}$ which is not in L although the lemma says it should. So, we obtain a contradiction... (do we?).

Why it is wrong:
Because the partition $x = b^{p-q}$, $y = a$ and $z = b^{p-1}$ is NOT the only partition possible. Maybe there are other partitions for which the lemma holds, but we don't know which they are. Hence, given w we must prove that no matter which partition is done (as long as it is done under the conditions of the lemma) we can find a new word $w' = xy^iz$ (ie. find a value i) such that $w' \notin L$.

Suggested Problems

Problem 1: Consider $L = \{ a^{3n} b^{4n} \ : \ n, m \geq 0 \}$. Is it regular or not? Prove it.

Problem 2: Consider $L = \{ a^n b a^{4n} \ : \ n > 0 \}$. Is it regular or not? Prove it.

Problem 3: Consider L="All strings in $\{ a, b, c \}^*$ that end with a palindrome of length 3". (*Palindromes* are the words that are read the same way from both ends, e.g. **atoyota**). Is it regular or not? Prove it.

Problem 4: Consider $L = \{ a^n b^m \ : \ n \neq m \}$. Prove this language is not regular by using both

1. Closure properties of regular languages (eg. "if L were regular, then $L \cap R$ would be regular because we know R is" or "$\overline{L}$ would be also" or "L^R would be also" or "$L \cup R$ would be also, etc.
2. Pumping Lemma. This is tricky, but it definitely shows how the contradiction should come from any word partition. *Hint: notice that most choices of words do not work. Use the word $a^p b^{p+p!}$, where $p! = p \cdot (p-1) \cdots 2 \cdot 1$; use the fact that any number k, $1 \leq k \leq p$ divides $p!$ exactly.*

The solutions for this problems will be posted next Wednesday Jan. 31.