Alin Tomescu, *6.840 Theory of Computation (Fall 2013)*, taught by Prof. Michael Sipser

Pumping lemmas

For regular languages

If A is regular, then $\exists p$ (the pumping length) such that $\forall s \in A$ where $|s| \geq p$, we can divide s into three pieces $s = xyz$, satisfying the following conditions:

(1) $xy^iz \in A, \forall i \geq 0$
(2) $|xy| \leq p$
(3) $y \neq \varepsilon$

Example: 0^n1^n is not regular

For context-free languages

If A is context-free, then $\exists p$ (the pumping length) such that $\forall s \in A$ where $|s| \geq p$, we can divide s into five pieces $s = uvxyz$, satisfying the following conditions:

(1) $uv^ixy^iz \in A, \forall i \geq 0$
(2) $|vxy| \leq p$
(3) $vy \neq \varepsilon$

Example: $a^nb^nc^n$ is not context-free. If it were, then let p be the pumping length and consider $w = a^pb^pc^p$. We can get a contradiction on $|vxy| \leq p$ by noticing that wherever it "falls" within w, if we pump $w = uvxyz$ up to uv^2xy^2z the resulting string is not in the language anymore due to imbalanced number of letters or out of order letters.

Notes on the pumping lemma

A language will have a **minimum pumping** length p. Also, all $p' > p$ will be valid pumping lengths.

- For finite DFAs, the minimum pumping length that vacuously satisfies the lemma conditions is k where k is the number of states.
- For infinite DFAs, the minimum pumping length is k, where k is the number of states.
- For finite grammars, *it seems* like a working pumping length is 2^m, where m is the # of variables in the CNF grammar. This is because you can only generate strings up to 2^{m-1} in size with m CNF grammar variables.

How do I think about using the conditions? For any p, EVERY string longer than p must have at least one way to divide it so that the conditions hold. Thus, when you are finding a contradiction, you need to find ONE string (there may be more, but you don't care about that) for which there is NO way to divide it up so that the conditions hold.

IMPORTANT: Once you've found the string, you only need to find one value of i for which the string cannot be pumped! Just one.