Pumping Lemmas

The main usefulness of the two pumping lemmas is to prove that a particular language is not regular, or context-free, as the case may be. Each lemma states that every language in the class has a certain property, and thus if we can prove that a given language L does not have that property, L is not in the class.

If w is a string and a is a symbol, we write $\#_a(w)$ to be the number of instances of the symbol a in the string w.

Lemma 1 (Pumping Lemma for Regular Languages) *If L is a regular language, there exists a positive integer p, called the pumping length of L, such that for any string $w \in L$ whose length is at least p, there exist strings x, y, z such that the following conditions hold.*

1. $w = xyz$
2. $|y| \geq 1$
3. $|xy| \leq p$
4. *for any* $i \geq 0$, $xy^i z \in L$.

Note that the the value of p is not unique: if p is a pumping length of L, so is every integer larger than p. Every regular language has a minimum pumping length.

Lemma 2 *A regular language accepted by an NFA with n states has pumping length n.*

Lemma 1 follows from Lemma 2. Do you see why?

Try proving Lemma 2 before looking at the proof on the next page.

Example

Let L be the language of all base 2 numerals for multiples of 5, where leading zeros are not allowed. The minimum number of states of any NFA which accepts L is six. Thus, by Lemma 2, L has pumping length 6. For example, if $w = 100011$, which means 35, let $x = 10$, $y = 001$, and $z = 1$. The first three conditions obviously hold. If we let $i = 0$, we get $xz = 101$, which means 5, while if $i = 2$ or $i = 3$, we get $xy^2z = 100010011$ which means 275, or $xy^3z = 100010010011$ which means 2115. The pumping length cannot be 4, since 1111, which means 15, does not have a pumpable substring.

Question: Does L have Pumping Length 5?

Proof: (Of Lemma 2.) Suppose M is an NFA which accepts L, and let $w \in L$ have length at least n. An accepting computation of w has length $|w|$, hence must visit some state q of M at least twice. That computation of w visits q after reading x, then visits q again after reading y, where y is not empty, and finally visits a final state after reading z, where $w = xyz$. For any $i \geq 0$, xy^iz has an accepting computation which visits q after reading x, then visits q i additional times while reading y^i, then visits a final state while reading z. $\blacksquare$

Lemma 3 (Pumping Lemma for Context-Free Languages) *If L is a context-free language, there exists a positive integer p, called the pumping length of L, such that for any string $w \in L$ whose length is at least p, there exist strings u, v, x, y, z such that the following conditions hold.*

1. $w = uvxyz$
2. $|v| + |y| \geq 1$
3. $|vxy| \leq p$
4. *for any* $i \geq 0$, $uv^ixy^iz \in L$.

Note that the the value of p is not unique: if p is a pumping length of L, so is every integer larger than p. There is a minimum pumping length.

Example

Let L be the language consisting of all palindromes over $\{a, b\}$. A string w is a palindrome if $w = w^R$, the reversal of w. The following is an unambiguous grammar for L:

$S- > aSa|bSb|a|b|\lambda$

What is the minimum pumping length of L?

L has pumping length 3. If a palindrome w has even length, the substring aa or bb in the middle of the string is pumpable. Without loss of generality, $w = taat^R$ for some string t. Without loss of generality. Let $u = t$, $v = a$, $x = \lambda$, $y = a$, and $z = t^R$. The first three conditions are obviously satisfied. For any $i \geq 0$, $uv^ixy^iz = ua^ia^iu^R \in L$.

If w has odd length, then there are four possibilities:
$w = taaat^R$
$w = tabat^R$
$w = tbabt^R$
$w = tbbbt^R$
In the first case, we let $u = t$, $v = a$, $x = a$, $y = a$, and $z = t^R$. In the second case, we let $u = t$, $v = a$, $x = b$, $y = a$, and $z = t^R$. In each case, four conditions are satisfied. The other two cases are similar.

Questions: Does L have pumping length 2? Does L have pumping length 1?

Using the Pumping Lemmas

Lemma 1 states a property that all regular languages have. Hence, if a language fails to satisfy that property, it is not regular. Similarly, if language fails to satisfy the property given by Lemma 3, it is not context-free.

Let $L_1 = \{a^n b^n : n \geq 0\}$, and let $L_2 = \{a^n b^n c^n : n \geq 0\}$. We use Lemma 1 to prove L_1 is not regular, and Lemma 3 to prove L_2 is not context-free.

Theorem 1 *L_1 is not regular.*

Proof: By contradiction. We assume L_1 is regular. Let p be a pumping length of L. (We usually say, *the* pumping length, despite the fact that it is not unique.) Let $w = a^p b^p$. Note that $|w| = 2p \geq p$, hence there exist strings x, y, z such that
1. $w = xyz$
2. $|xy| \leq p$
3. $|y| > 0$
4. For any $i \geq 0$, $xy^i z \in L_1$.

By 1. and 2., xy is a prefix of w of length no greater than p. Since the first p symbols of w are a's, that implies xy is a string of a's, hence y is also a string of a's. Write $y = a^j$. By 3., $j > 0$. Let $i = 0$. By 4., $xy^0 z = xz \in L_1$. But $xz = a^{p-j} b^p \notin L_1$ since $\#_a(xy \neq \#_b(xy)$, contradiction. ∎

Theorem 2 *L_2 is not context-free.*

Proof: By contradiction. We assume L_2 is context-free. Let p be the pumping length of L. Let $w = a^p b^p c^p$. Note that $|w| = 3p \geq p$, hence there exist strings x, y, z, u, v such that
1. $w = uvxyz$
2. $|vxy| \leq p$
3. $|v| + |y| > 0$
4. For any $i \geq 0$, $uv^i xy^i z \in L_2$.

Consider the frequency of each symbol in the substring vxy. We claim that either $\#_a(vxy) = 0$ or $\#_c(vxy) = 0$, since otherwise, vxy would contain some a and also some c in w, as well as all the symbols in between, which would give it a length of at least $p + 2$. But, by 2., $|vxy| \leq p$, contradiction.

Now, without loss of generality, $\#_a(vxy) = 0$. Then uxz, which is a member of L by 4., by choosing $i = 0$. Since $\#_a(v) = \#_a(y) = 0$, $\#_a(uxz) = p$. Each member of L has equal numbers of each of the three symbols, thus $|uxz| = 3p$. Since $|uvxyz| = 3p$, we have $|u| + |v| = 0$, which contradicts 3. ∎