Computer Science 351

Nonregular Languages, Part One

Instructor: Wayne Eberly

Department of Computer Science
University of Calgary

Lecture #9

Goal for Today

Goals for Today:

- Introduction of a theorem called the ***Pumping Lemma for Regular Languages***
- Describe how to use this to prove that a language $L \subseteq \Sigma^\star$ is ***not*** regular!

Pumping Lemma for Regular Languages

Pumping Lemma: Let Σ be an alphabet and let $A \subseteq \Sigma^{\star}$.

If A is a regular language, then there is a number $p \geq 1$ (called the ***pumping length*** for A) — which only depends on A — such that if s is any string in A with length at least p, then s can be divided into three pieces $s = xyz$ (for $x, y, z \in \Sigma^{\star}$), satisfying the following three conditions.

1. $xy^i z \in A$ for every integer i such that $i \geq 0$.
2. $|y| > 0$ (so that $y \neq \lambda$).
3. $|xy| \leq p$.

Note: y^i is the concatenation of i copies of the string y.

Pumping Lemma for Regular Languages

Sketch of Proof:

- Since A is regular there exists a deterministic finite automaton
$$M = (Q, \Sigma, \delta, q_0, F)$$
such that $L(M) = A$.
- Set p to be $|Q|$.
- The claim is "vacuously" satisfied if there are no strings in A with length at least p — so we may now assume that at least one such string exists.

Pumping Lemma for Regular Languages

Now let s be a string in A such that $|s| \geq p$.

- Consider the sequence of states

$$r_0, r_1, r_2, \ldots, r_p$$

  reached by processing the first p symbols in s (so that $r_0 = p_0$, M's start state).

- This is a sequence of states in Q with length $|Q| + 1$ — so some state $t \in Q$ must appear at least *twice*.
- Let a and b be the positions of the *first* and *second* occurrences of t in this sequence of states — so $0 \leq a < b \leq p$.

Pumping Lemma for Regular Languages

Now suppose that

$$s = \alpha_1 \alpha_2 \ldots \alpha_\ell$$

where $\ell = |s| \geq p$.

- Let $x = \alpha_1 \alpha_2 \ldots \alpha_a$ (so $a = 0$ and $x = \lambda$ if $t = q_0$). Then

$$\delta^\star(q_0, x) = t. \tag{1}$$

- Let $y = \alpha_{a+1} \alpha_{a+2} \ldots \alpha_b$, so that $|y| = b - a \geq 1$ (and $y \neq \lambda$, and

$$\delta^\star(t, y) = t. \tag{2}$$

- Let $z = \alpha_{b+1} \alpha_{b+2} \ldots \alpha_\ell$ — so that $s = xyz$ and

$$\delta^\star(t, z) = \delta^\star(q_0, s) \in F. \tag{3}$$

Pumping Lemma for Regular Languages

- The equations at lines (1), (2), and (3) can be used to show that
$$\delta^\star(xy^iz) = \delta(s) \in F$$
— so that $xy^iz \in A$ — for every non-negative integer i.
- It follows by the choice of x, y and z that $|y| > 0$ and $|xy| \leq p$.
- Since s was an "arbitrarily chosen" string such that $s \in A$ and $|s| \geq p$, this establishes the claim. □

Applying the Pumping Lemma

- We ***will not*** be using this theorem to try to prove that a given language A is a regular language.
- Instead, this will be used in proofs — by contradiction — to prove that various languages are *not* regular.

An Example

Let $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ and consider the language

$$A = \{\mathtt{a}^n\mathtt{b}^n \mid n \geq 0\}$$

so that A includes strings $\lambda, \mathtt{ab}, \mathtt{aabb}, \mathtt{aaabbb}, \ldots$, but A does not include $\mathtt{ba}$ or $\mathtt{abb}$.

Claim: A is not a regular language.

Proof: Suppose, ***to obtain a contradiction***, that A *is* a regular language.

An Example

Then it follows by the Pumping Lemma for Regular Languages, that there is a number $p \geq 1$ such that if s is any string in A with length at least p, then s can be divided in to three pieces $s = xyz$ (for $x, y, z \in \Sigma^\star$), satisfying the following three conditions.

1. $xy^iz \in A$ for every integer i such that $i \geq 0$.
2. $|y| > 0$ (so that $y \neq \lambda$).
3. $|xy| \leq p$.

An Example

Consider the string $s = \mathrm{a}^p\mathrm{b}^p$.

- $s \in A$, since $s = \mathrm{a}^n\mathrm{b}^n$ when $n = p$.
- $|s| = 2p \geq p$.

It follows that $s = xyz$ for strings $x, y, z \in \Sigma^\star$ that satisfy properties #1 – #3, as above.

- Since xy is a prefix of s with length at most p (by property #3), and the first p symbols in s are all a's, $xy = \mathrm{a}^k$ for some integer k such that $0 \leq k \leq p$.

  Since $s = \mathrm{a}^p\mathrm{b}^p = xyz = a^k z$, it follows that $z = a^{p-k}b^p$.

An Example

- By property #2, $|y| > 0$ so that $y \neq \lambda$. Thus (since $|xy| = |x| + |y| = k$), $|x| = h$ and $|y| = \ell$ for integers h and ℓ such that $h \geq 0$, $\ell \geq 1$, and $h + \ell = k$.

  Since $xy = \mathrm{a}^k$ it now follows that $x = \mathrm{a}^h$ and $y = \mathrm{a}^\ell$.

- Let $i = 0$. Then

  $$xy^iz = x\lambda z = xz = \mathrm{a}^h\mathrm{a}^{p-k}\mathrm{b}^p = \mathrm{a}^{p-\ell}\mathrm{b}^p$$

  since $h + \ell = k$. Thus $xy^iz \notin A$, since $\ell > 0$ (so that $p - \ell \neq p$).

- Thus property #1 is ***not*** satisfied (since $xy^iz \notin A$ when $i = 0$), and a ***contradiction*** has been obtained.

- It follows that A is not a regular language. □

Summary of Process

In order to prove that a given language $A \subseteq \Sigma^\star$ is ***not*** a regular language, using the Pumping Lemma for Regular Languages, you should do the following things.

1. Assume — to obtain a contradiction — that A *is* a regular language.

   Note: It is a "courtesy to the reader" to *say* that you are giving a proof by contradiction, at the beginning of your proof.

Summary of Process

2. Note that it follows, by the Pumping Lemma for Regular Languages, that there exists a number $p \geq 1$ such that if s is any string in A with length at least p, then s can be divided in to three pieces $s = xyz$ (for $x, y, z \in \Sigma^{\star}$), satisfying the following three conditions.
    1. $xy^iz \in A$ for every integer i such that $i \geq 0$.
    2. $|y| > 0$ (so that $y \neq \lambda$).
    3. $|xy| \leq p$.

Summary of Process

Things to Note:

- You ***cannot*** pick and use a particular value for p, or assume anything more about its value: You would be introducing *another* assumption if you did that, and then you could not conclude that the *original* assumption was incorrect, once a contradiction was obtained.
- It is advisable to *state* the properties that follows, by the Pumping Lemma, in your written proof — because you will need to refer to all of them, later on in the proof.

Summary of Process

3. Describe a string s — that will generally depend on the value p that has just been chosen. Give a (generally short) proof that $s \in A$ and $|s| \geq p$.

   Note: You ***do*** get to pick the string s that will be used for the rest of this proof!

   However, this is generally the trickiest part of this process! Some choices of s will make the rest of this easy. Other choices will make it impossible to correctly complete the proof.

Summary of Process

4. Observe that it now follows that there exist strings $x, y, z \in \Sigma^\star$ such that $s = xyz$ and properties #1, #2, and #3 (listed in the "Pumping Lemma for Regular Languages") hold.

   Note: Once again, you ***do not*** get to choose the strings x, y and z to work with — you need to prove that there is a problem with ***all*** possible choices of these strings.

5. Show that if x and y are ***any*** initial strings of s such that properties #2 and #3 hold, then $xy^iz \notin A$ for ***some*** nonnegative integer i.

6. Note that a ***contradiction*** has now been obtained, so the assumption must be false: A is *not* a regular language.