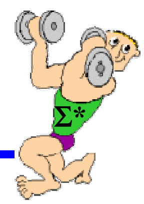
Beyond the Regular world…

- Are there languages that are *not* regular?
- **Idea:** If a language violates a property obeyed by all regular languages, it cannot be regular!
  - **Pumping Lemma** for showing *non-regularity* of languages

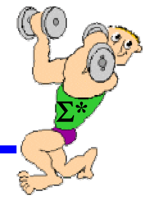
http://www.ipjnet.com/schwarzenegger2/pages/arnold_01.htm

The Pumping Lemma for Regular Languages



- **What is it?**
  - A statement ("lemma") that is true for all regular languages
- **Why is it useful?**
  - Can be used to show that certain languages are *not regular*
  - <u>How?</u> *By contradiction*: Assume the given language is regular and show that it does not satisfy the pumping lemma

More about the Pumping Lemma



- **What is the idea behind it?**
  - Any regular language L has a DFA M that recognizes it
  - If M has **p states** and accepts a string of length ≥ p, the sequence of states M goes through must contain a **cycle** (repetition of a state) due to the *pigeonhole principle*! Thus:
  - *All strings* that make M go through this cycle 0 or any number of times are also accepted by M and *should be in L.*

Formal Statement of the Pumping Lemma

- **Pumping Lemma**: If L is regular, then ∃ p such that ∀ s in L with $|s| \geq p$, ∃ x, y, z with $s = xyz$ and:
  1. $xy^iz \in L \; \forall \; i \geq 0$, and
  2. $|y| \geq 1$, and
  3. $|xy| \leq p$.
- Proof on board…(also in the textbook)
- Proved in 1961 by Bar-Hillel, Peries and Shamir

Pumping Lemma in Plain English

- Let L be a regular language and let p = "pumping length" = no. of states of a DFA accepting L
- Then, any string s in L of length ≥ p can be expressed as $s = xyz$ where:
  - y is not empty (y is the cycle)
  - $|xy| \leq p$ (cycle occurs within p state transitions), and
  - any "pumped" string xy^iz is also in L for all $i \geq 0$ (go through the cycle 0 or more times)

Using The Pumping Lemma

- In-Class Examples: Using the pumping lemma to show a language L is *not regular*
  - 5 steps for a proof by contradiction:

1. Assume L is regular.
2. Let p be the pumping length given by the pumping lemma.
3. Choose cleverly an s in L of length at least p, such that
4. For *all ways* of decomposing s into xyz, where $|xy| \leq p$ and y is not null,
5. There is an $i \geq 0$ such that xy^iz is not in L.

Proving non-regularity as a Two-Person game

- An alternate view: Think of it as a *game between you and an opponent (JC)*:
  1. **You**: Assume L is regular
  2. **JC**: Chooses some value p
  3. **You**: Choose cleverly an s in L of length $\geq$ p
  4. **JC**: Breaks s into some xyz, where $|xy| \leq$ p and y is not null,
  5. **You**: Need to choose an $i \geq 0$ such that xy^iz is not in L (in order to win (the prize of non-regularity)!)

  (Note: Your i should work for all xyz that JC chooses, given your s)

Proving Non-Regularity using the Pumping Lemma

- Examples: Show the following are not regular
  - $L_1 = \{0^n1^n \mid n \geq 0\}$ over the alphabet {0, 1}
  - L_2 = {w | w contains equal number of 0s and 1s} over the alphabet {0, 1}
  - ADD = {a=b+c | a, b, c are binary numbers and a is the sum of b and c} over the alphabet {0, 1, =, +}
  - PRIMES = $\{0^n \mid n$ is prime$\}$ over the alphabet {0}

Da Pumpin' Lemma

(Orig. lyrics: Harry Mairson)

Any regular language *L* has a magic numba *p*
And any long-enuff word *s* in *L* has da followin' propa'ty:
Amongst its first *p* symbols is a segment you can find
Whoz repetition or omission leaves *s* amongst its kind.

So if ya find a language *L* which fails dis acid test,
And some long word ya pump becomes distinct from all da rest,
By contradiction you have shown dat language *L* is not
A regular homie, resilient to the damage you've caused.

But if, upon the other hand, *s* stays within its *L*,
Then either *L* is regulah, or else you chose not well.
For *s* is *xyz*, where *y* cannot be empty,
And *y* must come before da $p+1^{th}$ symbol is read.

Based on: http://www.cs.brandeis.edu/~mairson/poems/node1.html

If $\{0^n1^n \mid n \geq 0\}$ is not Regular, what is it?

Enter…the world of Grammars