
The Pumping Lemma & Closure Properties

Martin Fränze

Informatics and Mathematical Modelling
The Technical University of Denmark

02140 Languages and Parsing, MF, Fall 2004 – p.1/8

What you'll learn

1. Repeatable sub-strings:

- What's "regular" about regular languages

2. The pumping lemma:

- Shows that repeatable sub-strings are ubiquitous in infinite regular languages

3. How to prove languages being non-regular:

- An application of the pumping lemma

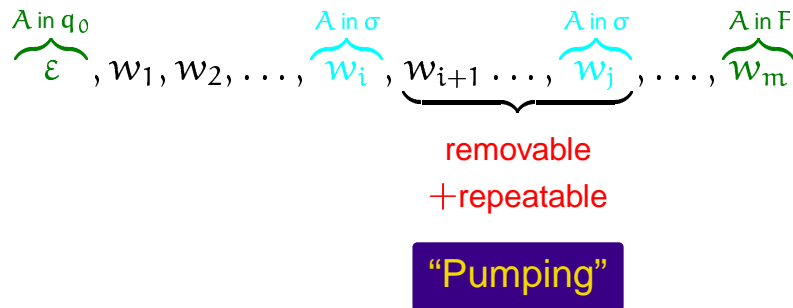
4. Some closure properties of the class of regular languages

02140 Languages and Parsing, MF, Fall 2004 – p.2/8

Regularity

Be A a DFA with n states,
and $w \in L(A)$ with $|w| \geq n$

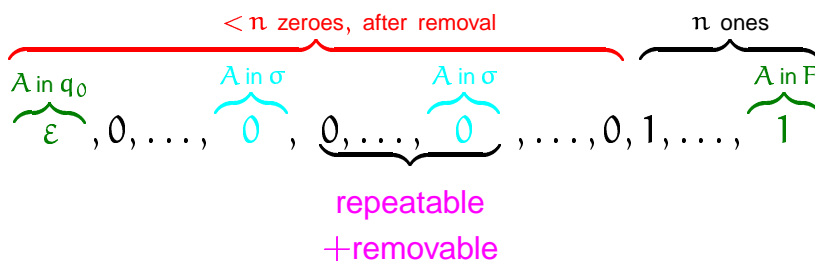
A traversing w :



Non-regularity

Assume, $L = \{0^k 1^k \mid k \in \mathbb{N}\}$ were regular, being accepted by an n -state DFA:

A traversing $0^n 1^n$:



- Shows that any DFA recognizing all strings in L recognizes further strings outside L .

$\Rightarrow L = \{0^k 1^k \mid k \in \mathbb{N}\}$ can't be recognized by finite automata!

The pumping lemma for regular languages

Thm: For each *regular* language L there is a constant $n \in \mathbb{N}$ such that each $w \in L$ with $|w| \geq n$ can be split into $w = xyz$ such that

1. $y \neq \varepsilon$,
2. $|xy| \leq n$,
3. $xy^kz \in L$ for *each* $k \in \mathbb{N}$.

N.B.: Doesn't imply that *all* regular languages are pumpable!
All *infinite* ones are, however.

Main use of pumping lemma is to prove non-regularity of some L !

Non-regularity — proof recipe

In order to **show L to be non-regular**, proceed as follows:

1. Take $n \in \mathbb{N}$ **arbitrarily** (i.e., no constraints to be imposed from your side);
2. Provide construction rules — obviously dependent on n — for the a word $w \in L$ with $|w| \geq n$.
3. Take an arbitrary split of w into $w = xyz$ satisfying
 - (a) $y \neq \varepsilon$, and
 - (b) $|xy| \leq n$.I.e., (a) and (b) are the only properties enforced.
4. Provide a construction rule for $k \in \mathbb{N}$ such that $x(y^k)z \notin L$.
5. Prove that your constructions satisfy *all* their claimed properties ($w \in L$, $|w| \geq n$, $x(y^k)z \notin L$), unless obvious.

Examples

Prove the following languages to be non-regular:

- $L = \{w \in \{0, 1\}^* \mid \#_0(w) \leq \#_1(w)\}$
- For any $n \in \mathbb{N}$, take $w = 0^n 1^n$, then pump.
- $L = \{w \in \{0\}^* \mid |w| \text{ is prime}\}$
- For any $n \in \mathbb{N}$, take $w = 0^p$ for a prime $p \geq n + 2$ such that $|z| \geq 2$ in the split $xyz = w$. Then pump to $x(y)^{|xz|}z$.
 $|x(y)^{|xz|}z| = \underbrace{(|y| + 1)}_{\geq 2} \underbrace{(|xz|)}_{\geq 2}$ is not prime.

Closure properties of regular languages

The regular languages are closed under

- **union** — i.e., the union of two regular languages is regular,
- **intersection** — i.e., the intersection of two reg. lang.s is regular,
- **complement** — i.e., the complement of a reg. lang. is regular,
- **difference** — i.e., the difference of two reg. lang.s is regular,
- **reversal** — i.e., the language containing the reversals of the words of a reg. lang. is regular,
- **closure (Kleene star)** — i.e., the closure of a reg. lang. is regular,
- **concatenation** — i.e., the catenation of two reg. lang.s is regular,
- **homomorphism** — i.e., the image of a reg. lang. under a homomorphism is regular,
- **inverse homomorphism** — i.e., the pre-image of a reg. lang. under a homomorphism is regular.