Discrete Mathematics

(Not only) Regular Languages — Spring 2025

Konstantin Chukharev

§1 Regular Languages

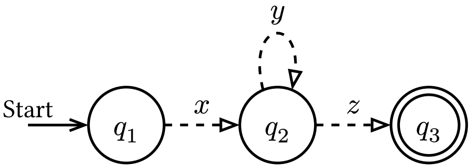
Regular Expressions

Regular languages can be composed from "smaller" regular languages.

- Atomic regular expressions:
  - $\varnothing$, an empty language
  - ε, a singleton language consisting of a single ε word
  - $\mathtt{a}$, a singleton language consisting of a single 1-letter word a, for each $a \in \Sigma$
- Compound regular expressions:
  - $R_1 R_2$, the concatenation of R_1 and R_2
  - $R_1 \mid R_2$, the union of R_1 and R_2
  - $R^* = RRR\ldots$, the Kleene star of R
  - (R), just a bracketed expression
  - Operator precedence: `ab*c|d` $\triangleq ((\mathtt{a}\ (\mathtt{b}^*))\ \mathtt{c}) \mid \mathtt{d}$

Re-visiting States

- Let D be a DFA with n states.
- Any string w accepted by D that has length at least n must visit some state twice.
- Number of states visited is equal to $|w| + 1$.
- By the pigeonhole principle, some state is "duplicated", i.e. visited more than once.
- The substring of w between those *revisited states* can be removed, duplicated, tripled, etc. without changing the fact that D accepts w.



Informally:

- Let L be a regular language.
- If we have a string $w \in L$ that is "sufficiently long", then we can *split* the string into *three pieces* and *"pump"* the middle.

Re-visiting States [2]

- We can write $w = xyz$ such that xy^0z, xy^1z, xy^2z, ..., xy^nz, ... are all in L.
  - Notation: y^n means "n copies of y".

Weak Pumping Lemma

Theorem 1 (Weak Pumping Lemma for Regular Languages):

- For any regular language L,
  - There exists a positive natural number n (also called *pumping length*) such that
    - For any $w \in L$ with $|w| \geq n$,
      - There exists strings x, y, z such that
        - For any natural number i,
          - $w = xyz$ (w can be broken into three pieces)
          - $y \neq \varepsilon$ (the middle part is not empty)
          - $xy^iz \in L$ (the middle part can repeated any number of times)

Example: Let $\Sigma = \{0, 1\}$ and $L = \{w \in \Sigma^* \mid w \text{ contains } 00 \text{ as a substring}\}$. Any string of length 3 or greater can be split into three parts, the second of which can be "pumped".

Example: Let $\Sigma = \{0, 1\}$ and $L = \{\varepsilon, 0, 1, 00, 01, 10, 11\}$. The weak pumping lemma still holds for finite languages, because the pumping length n can be longer than the longest word in the language!

Testing Equality

Definition 1: The *equality problem* is, given two strings x and y, to decide whether $x = y$.

Example: Let $\Sigma = \{0, 1, \#\}$. We can *encode* the equality problem as a string of the form $x\#y$.
- "Is *001* equal to *110* ?" would be $001\#110$.
- "Is *11* equal to *11* ?" would be $11\#11$.
- "Is *110* equal to *110* ?" would be $110\#110$.

Let EQUAL $= \{w\#w \mid w \in \{0, 1\}^*\}$.

Question: Is EQUAL a *regular* language?

A typical word in EQUAL looks like this: 001#001.
- If the "middle" piece is just a symbol $\#$, then observe that 001 001 $\notin$ EQUAL.
- If the "middle" piece is either completely to the left or completely to the right of $\#$, then observe that any duplication or removal of this piece is not in EQUAL.
- If the "middle" piece includes $\#$ and any symbols from the left/right of it, then, again, observe that any duplication or removal of this piece is not in EQUAL.

Testing Equality [2]

Theorem 2: EQUAL is not a regular language.

Proof: By contradiction. Assume that EQUAL is a regular language.

Let n be the pumping length guaranteed by the weak pumping lemma. Let $w = 0^n\#0^n$, which is in EQUAL and $|w| = 2n + 1 \geq n$. By the weak pumping lemma, we can write $w = xyz$ such that $y \neq \varepsilon$ and for any $i \in \mathbb{N}$, $xy^i\#z \in$ EQUAL. Then y cannot contain $\#$, since otherwise if we let $i = 0$, then $xy^0\#z = x\#z$ does not contain $\#$ and would not be in EQUAL. So y is either completely to the left of $\#$ or completely to the right of $\#$.

Let $|y| = k$, so $k > 0$. Since y is completely to the left or right of $\#$, then $y = 0^k$.

Now, we consider two cases:
Case 1: y is to the left of $\#$. Then $xy^2z = 0^{n+k}\#0^n \notin$ EQUAL, contradicting the weak pumping lemma.
Case 2: y is to the right of $\#$. Then $xy^2z = 0^n\#0^{n+k} \notin$ EQUAL, contradicting the weak pumping lemma.

In either case we reach a contradiction, so our assumption was wrong. Thus EQUAL is not regular. $\square$

Non-regular Languages

- The weak pumping lemma describes a property common to *all* regular languages.
- Any language L which does not have this property *cannot be regular*.
- What other languages can we find that are not regular?

Example: Consider the language $L = \{0^n 1^n \mid n \in \mathbb{N}\}$.

- $L = \{\varepsilon, 01, 0011, 000111, 00001111, \ldots\}$
- L is a classic example of a non-regular language.
- **Intuitively:** if you have *only finitely many states* in a DFA, you cannot *"remember"* an arbitrary number of 0s to match *the same* number of 1s.

How would we prove that L is non-regular?

Pumping Lemma as a Game

The weak pumping lemma can be thought of as a *game* between **you** and an **adversary**.

- **You win** if you can prove that the pumping lemma *fails*.
- **The adversary wins** if the adversary can make a choice for which the pumping lemma *succeeds*.

The game goes as follows:

- The adversary chooses a pumping length n.
- You choose a string w with $|w| \geq n$ and $w \in L$.
- The adversary breaks it into x, y, and z.
- You choose an i such that $xy^iz \notin L$ *(if you can't, you lose!)*.

Pumping Lemma as a Game [2]

$$L = \{0^n 1^n \mid n \in \mathbb{N}\}$$

Adversary	You
Maliciously choose pumping length n	
	Cleverly choose a string $w \in L$, $\lvert w \rvert \geq n$
Maliciously split $w = xyz$, $y \neq \varepsilon$	
	Cleverly choose an i such that $xy^i z \notin L$
Lose	Win

$\{0^n 1^n\}$ is not regular

Pumping Lemma as a Game [3]

Theorem 3: $L = \{0^n 1^n \mid n \in \mathbb{N}\}$ is not regular.

Proof: By contradiction. Assume that L is regular.

Let n be the pumping length guaranteed by the weak pumping lemma. Consider the string $w = 0^n 1^n$. Then $|w| = 2n \geq n$ and $w \in L$, so we can write $w = xyz$ such that $y \neq \varepsilon$ and for any $i \in \mathbb{N}$, we have $xy^i z \in L$.

We consider three cases:
Case 1: y consists solely of 0s. Then $xy^0 z = xz = 0^n - |y| 1^n$, and since $|y| > 0$, $xz \notin L$.
Case 2: y consists solely of 1s. Then $xy^0 z = xz = 0^n 1^n - |y|$, and since $|y| > 0$, $xz \notin L$.
Case 3: y consists of $k > 0$ 0s followed by $m > 0$ 1s. Then $xy^2 z = 0^n 1^m 0^k 1^n$, so $xy^2 z \notin L$.

In all three cases we reach a contradiction, so our assumption was wrong and L is not regular. □