Théorie des Langages Rationnels

The Pumping Lemma

Adrien Pommellet, LRE

March 29, 2023

Theorem

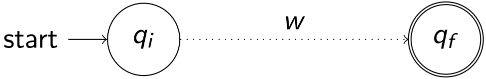
Rational languages are ***decidable****.*

Proof. Given a rational language $L \in \mathrm{Rat}_\Sigma$, there exists a DFA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}) = L$. And for any word $w \in \Sigma^*$ we can decide whether $\mathcal{A}$ accepts a word or not. Thus L is decidable.

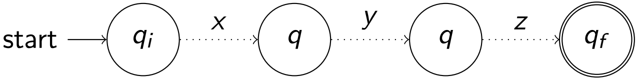
Do **non-rational yet decidable** languages exist?

The Pumping Lemma
A word too long

What if a DFA $\mathcal{A}$ with n states accepts a word w of length $|w| > n$?



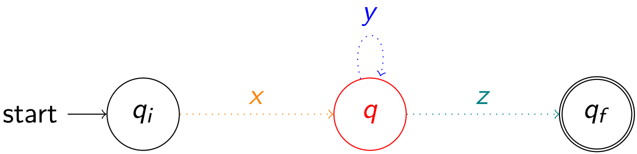
By the **pigeonhole principle**, w must visit a state q of $\mathcal{A}$ **at least twice**, hence the following path such that $w = x \cdot y \cdot z$ and $y \neq \varepsilon$.



The Pumping Lemma

Finding a loop

There is therefore a **loop** in $\mathcal{A}$ of which q is part.



As a consequence, $\mathcal{A}$ also accepts $x \cdot z$, $x \cdot y^2 \cdot z$, $x \cdot y^3 \cdot z$, and any word that follows the regular pattern $x \cdot y^* \cdot z$.

This property of **automata** can be generalized to **rational languages**:

Lemma (Pumping lemma)

Given a rational language $L \in \mathrm{Rat}_{\Sigma}$*, there exists a* ***pumping threshold*** $n_0 \in \mathbb{N}^*$ *such that for any word* $w \in L$ *of length* $n \geq n_0$*, there exist three words* $x, y, z \in \Sigma^*$ *such that* $w = x \cdot y \cdot z$*,* $y \neq \varepsilon$*, and* $\mathcal{L}(x \cdot y^* \cdot z) \subseteq L$*.*

Proof. It has been proven that there exists a DFA $\mathcal{A}$ such that $\mathcal{L}(\mathcal{A}) = L$. If $\mathcal{A}$ has n states, then any $n_0 > n$ is a suitable pumping threshold.

Non-rational Languages

A long sought-after example

Theorem

*The language $L = \{a^n b^n \mid n \in \mathbb{N}\}$ on the alphabet $\Sigma = \{a, b\}$ is **decidable** but **not rational**.*

Proof. We will prove that L is not rational by **contradiction**. Let us assume that L is rational.

Then the pumping lemma applies to L.

Let n_0 be a pumping threshold of L and consider $w = a^{n_0} b^{n_0}$. Since w is of length greater than n_0, it can be iterated upon.

Thus, there exist three words $x, y, z \in \Sigma^*$ such that $w = x \cdot y \cdot z$, $y \neq \varepsilon$, and $\mathcal{L}(x \cdot y^* \cdot z) \subseteq L$.

Now consider the three following cases:

1. There are more a's than b's in y. $aaabbb$
2. There are as many a's as bs in y. $aaabbb$
3. There are more b's than a's in y. $aaabbb$

Non-rational Languages

Handling the first case

Let us assume that there are more a's than b's in y: y is then of the form $y = a^u b^v$ where $u > v$.

Since $w = x \cdot y \cdot z \in L$, $|w|_a = |w|_b$, hence:

$$|x|_a + u + |z|_a = |x|_b + v + |z|_b$$

However, by the pumping lemma, $w' = x \cdot y^2 \cdot z \in L$. Thus:

$$\begin{gathered} |w'|_a = |w'|_b \\ |x|_a + 2u + |z|_a = |x|_b + 2v + |z|_b \\ u = v \end{gathered}$$

But $u > v$ and there is a **contradiction**.

Intuitively, if $w = a \cdot aa \cdot bbb$, then $w' = a \cdot aa \cdot aa \cdot bbb$ has 5 a's and 3 b's, hence is obviously not in L.

The third case is obviously similar to the first case.

Let us assume that there are as many a's as bs in y: y is then of the form $y = a^u b^u$.

By the pumping lemma, $w' = x \cdot y^2 \cdot z = x \cdot a^u b^u \cdot a^u b^u \cdot z \in L$.

But a a can't follow a b in a word of L and there is a **contradiction**.

The three cases all lead to a contradiction. Thus L can't be rational.

Practical Application

Exercise 1. Prove that the language $L = \{a^n b^n \mid n \in \mathbb{N}\}$ on the alphabet $\Sigma = \{a, b\}$ is **decidable**.

Answer

Consider the following algorithm on an input $w \in \Sigma^*$ that **decides** L:

```
def algo(w):
  if (len(w) == 0):
    return true
  i, n = 0
  while (w[i] == a):
    i, n = i+1, n+1
  while (w[i] == b):
    i, n = i+1, n-1
  if (i == len(w)) and (n == 0):
    return true
  else:
    return false
```

Non-rational Languages

The intuition

A rational language is recognized by an algorithm that uses **a constant amount of memory** that **does not depend on the size of the input**.

Each **state** of the DFA represents a possible configuration of the memory.

The language L here requires a **counter** `n` that can take unbounded values, hence can't be rational.

A language may follow the pumping lemma yet **not be rational**!

Practical Application

Exercise 2. Prove that the language $L_1 = \{n \in \mathbb{N} \mid 3 \text{ divides } n\}$ on the alphabet $\Sigma = \{0, \ldots, 9\}$ is **rational**.

Answer I

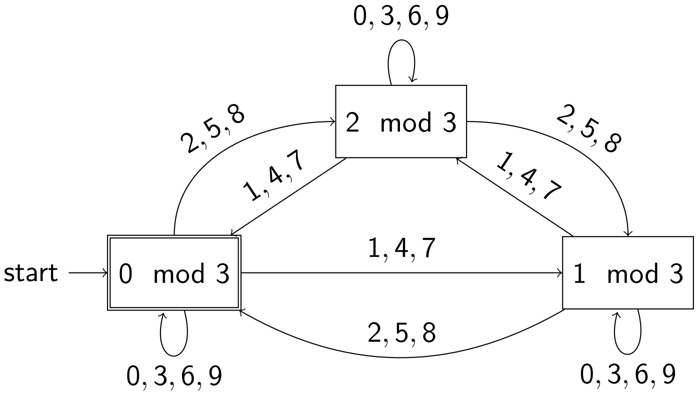
An integer is a multiple of three if and only if the **sum of his digits** is a multiple of 3 as well.

Our intuition is therefore to read the input integer from left to right, updating the **sum modulo 3** of its digits each time a digit is read.

We will need to design a DFA with only **three** states: one for each possible remainder of the sum.

Answer II

The following DFA accepts L_1:



Practical Application

Exercise 3. Prove that the language L_2 of words on the alphabet $\Sigma = \{a, b\}$ that have an even number of a's and an odd number of b's is **rational**.

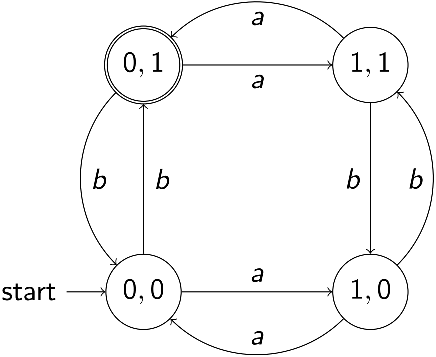
Answer I

In a similar fashion, we keep count of the number of a's modulo 2 and the number of b's modulo 2 using **two different bits**.

We will need to design a DFA with only **four** states: we encode on two bits relevant information kept in memory.

Answer II

The following DFA accepts L_2:



See you next class!