Daily Problem: 29–Jan–2026

Problem Statement

Prove the pumping lemma:

Let L be a regular language. Then there exists a constant $p \geq 1$ such that every word $w \in L$ with $|w| \geq p$ can be written as

$$w = xyz$$

where

$$|xy| \leq p, \qquad |y| \geq 1,$$

and for all integers $k \geq 0$,

$$xy^k z \in L.$$

Proof

Since L is regular, there exists a DFA

$$M = (Q, \Sigma, \delta, q_0, F)$$

such that $L = L(M)$. Let $|Q| = p$. We will show that this p works as the pumping length.

Step 1: Run the DFA on a long accepted string

Let $w \in L$ with $|w| \geq p$. Write

$$w = a_1 a_2 \cdots a_m$$

where each $a_i \in \Sigma$ and $m = |w| \geq p$.

Consider the sequence of states that the DFA visits while reading w:

$$r_0, r_1, r_2, \ldots, r_m,$$

where

$$r_0 = q_0, \qquad r_i = \delta(r_{i-1}, a_i) \text{ for } i = 1, \ldots, m.$$

Since $w \in L$, the run ends in an accepting state:

$$r_m \in F.$$

Step 2: Apply the pigeonhole principle

Look at the first $p + 1$ states in this run:

$$r_0, r_1, \ldots, r_p.$$

There are $p + 1$ states listed but only p states available in Q. Therefore, by the pigeonhole principle, two of these states must be equal. That is, there exist indices

$$0 \leq i < j \leq p$$

such that

$$r_i = r_j.$$

Step 3: Define x, y, z from the repetition

Define a decomposition $w = xyz$ as follows:

- Let $x = a_1 a_2 \cdots a_i$ (the prefix of length i),
- Let $y = a_{i+1} a_{i+2} \cdots a_j$ (the substring from $i + 1$ to j),
- Let $z = a_{j+1} a_{j+2} \cdots a_m$ (the remaining suffix).

Then clearly $w = xyz$.

Now verify the length conditions:

$$|xy| = j \leq p$$

since $j \leq p$ by construction, and

$$|y| = j - i \geq 1$$

because $i < j$.

Step 4: Pumping works

We claim that for every $k \geq 0$, the string xy^kz is also accepted by M, hence belongs to L.

To see why, observe what y does to the DFA state. Starting at state r_i, reading y takes the DFA to state r_j. But we chose $r_i = r_j$, so reading y returns the DFA to the same state:

$$\delta(r_i, y) = r_i.$$

Therefore, reading y any number of times keeps the DFA in state r_i:

$$\delta(r_i, y^k) = r_i \quad \text{for all } k \geq 0.$$

Now consider the run on xy^kz:

- reading x takes q_0 to r_i,
- reading y^k keeps the machine at r_i,
- reading z takes r_i to the same state that z took r_j to in the original accepting run.

Since the original run on $xyz = w$ ended in an accepting state $r_m \in F$, the run on xy^kz ends in the same accepting state. Hence

$$xy^kz \in L(M) = L \quad \text{for all } k \geq 0.$$

Conclusion

We have shown that with $p = |Q|$, every $w \in L$ with $|w| \geq p$ can be written as $w = xyz$ such that:

$$|xy| \leq p, \quad |y| \geq 1, \quad \text{and} \quad xy^kz \in L \text{ for all } k \geq 0.$$

This proves the pumping lemma for regular languages.