

Pumping Lemma for Finite Automata

Theorem

Let L be an infinite regular language. Let p be the number of states in a deterministic finite automaton that recognises L . Then any string s in L of length at least p can be broken into three substrings $s = xyz$ such that:

- $|y| > 0$,
- $|xy| \leq p$, and
- $xy^iz \in L$ for all $i \in \mathbb{N}_0$.

Rationale

Once we read p characters from s , we must have visited some state twice¹. Suppose q is a state we visit twice, and call y the substring of s that we read between the two visits.

When we can delete y from s and the automaton must accept this new string also. Likewise we can repeat y any number of times to create a new string that must also be accepted.

Note the automaton essentially forgets the path it took to a given state – once it arrives at a given state it can't remember how it got there.

Example

Non-regular example

$$L = \{0^i 1^i \mid i \in \mathbb{N}_0\}$$

$$s = 0^p 1^p = xyz$$

$$|xy| \leq p \Rightarrow y = 0^n, n \in \mathbb{N}$$

$$\Rightarrow 0^{p-n} 0^{2n} 1^n \in L$$

That's a contradiction.

References

¹At least one state at least twice.