Pumping Lemma for Regular Languages

26 January 2021 19:43

What are the limitations of DFAs / NFAs / regular languages? Which languages cannot be recognized by DFA / NFAs?.

PUMPING LEMMA: Gives a necessary condition for a language to be regular.

A is regular $\Rightarrow$ A can be pumped

$\not\Leftarrow$ The converse is not true.

The above implies that if A cannot be pumped, then A is not regular. This gives a condition for testing non regular languages. This is not a test for regular languages.

$\{\varepsilon, 01, 0011, 000111, 00001111, \ldots\}$

Example: $B = \{0^n 1^n \mid n \geq 0\}$. A DFA, in order to recognize B, in some sense, has to count the number of 0's. But this is not a bounded quantity. So it cannot be done with a limited number of states.

Intuition Only!

Pumping lemma gives us a way to formalize this intuition.

this intuition.

Was discovered by Michael Rabin and Dana Scott (in 1959) and later by Bar-Hillel, Perles and Shamir (1961).

<u>Statement</u>: If A is a regular language, then there exists a number p (pumping length) such that, if s is any string in A of length at least p, then s can be divided into three pieces $s = xyz$, such that

(1) for each $i \geq 0$, $xy^iz \in A$.

(2) $|y| > 0$

(3) $|xy| \leq p$.

<u>Proof</u>: (Intuition and Sketch) For full details see the book.

Let $M = (Q, \Sigma, \delta, q_1, F)$ be a DFA that recognizes A. We will show that pumping length $p = |Q|$ (number of states).

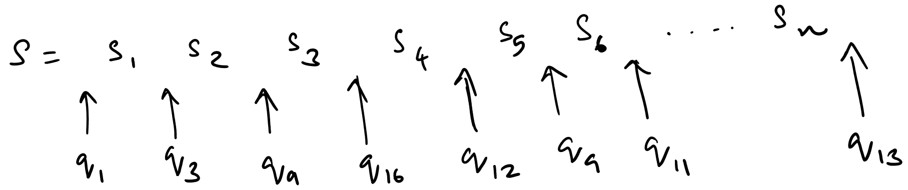
* What if all strings $s \in A$ are of length $< p$?

... holds vacuously.

* What if ...

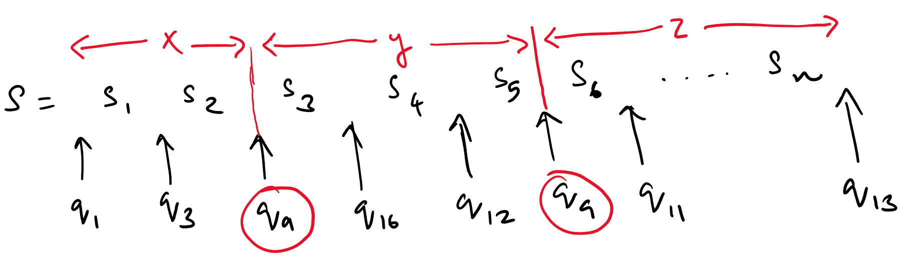
Then lemma holds vacuously.

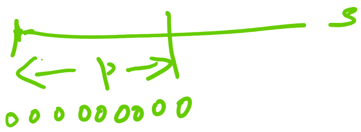
* For strings $s \in A$ with $|s| \geq p$, we will use an argument based on the pigeonhole principle.



Consider the sequence of states that M goes through while processing s. This starts with q_1, then say $q_3 \ldots$ etc. till say q_{13}.

If $|s| = n$, then the sequence has $n+1$ states, with some possible repetitions. When $|s| = n \geq p$, by pigeonhole principle, there exists at least one repeated state, since $n+1 \geq p+1 > p = |Q|$.





In the above, q_9 is the repeated state. Let

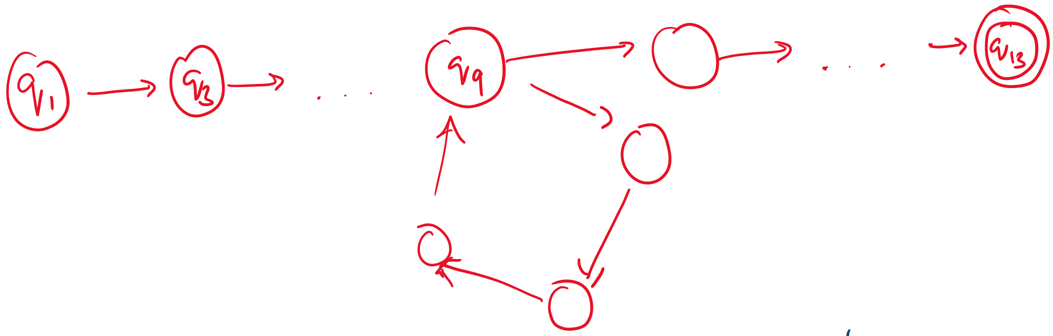
$x =$ s till the first occurrence of q_9

... ft. first to the second occurrence

$x =$ s till the first ...

$y =$ from the first to the second occurrence of q_9

$z =$ from the second occurrence of q_9 till the end of s.



In the DFA M, x takes M from q_1 to q_9

y takes M from q_9 to itself

z takes M from q_9 to q_{13}.

Consider xyyz. This will also be accepted. The difference is that there are two rounds of y instead of one. Similarly for xy^3z and xz.

Hence xy^iz is accepted by M and hence is in A for all $i \geq 0$.

Since there are two occurrences of at least one state (q_9 here in the above example), there exists a non empty string that is processed in between. Thus $|y| > 0$. (or $y \neq \epsilon$).

in between. Thus $|y| > 0$. (or $y \neq \epsilon$).

Pigeonhole principle guarantees that the first repetition occurs on or before s_p is processed.

Hence $|xy| \leq p$.

Read the formal proof from the book.