Lecture 19

Thursday, March 7, 2024 11:25 AM

Theorem. Assume M_1 and M_2 are DFAs and $L(M_1) = L(M_2)$. Assume M_1' and M_2' are the minimized (collapsed) versions of M_1 and M_2 respectively. Then M_1' and M_2' are isomorphic.

We skip the proof (Myhill-Nerode relations)

How can we check if two DFAs (M_1 and M_2) represent the same language?

* minimize them and then check for isomorphism.
* Time complexity?

Question: write a program that searches for all occurances of a substring w in string x.

$x = a\,b\,c\,\underbrace{a\,b\,\overbrace{a}\,b\,a}\,c\,c\,a\,b\,c\,\underbrace{a\,b\,a}\,c\,c$

$w = aba$

Assume $|x| = m$, $|w| = n$.

* Approach 1: check whether w occurs in each location in x.

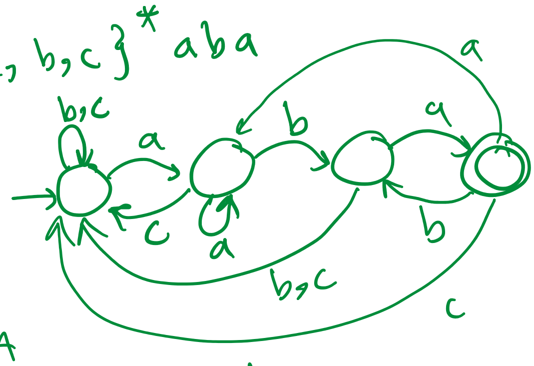
Runtime: $\Theta(n \cdot m)$

* Approach 2: use DFAs!

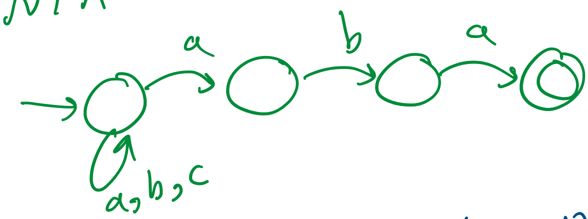
 * Build a DFA for $\Sigma^* . w = @w$

* Build a DFA for $\Sigma^* \cdot \omega = @\omega$
* Feed x to the DFA.
* Whenever the DFA is in an accept state we have found an occurrance of ω.

$\{a, b, c\}^* aba$



NFA



cost of running the DFA on input x: $\Theta(|x|) = \Theta(m)$

cost of building the DFA:

* create an NFA and use subset construction to find a DFA: $\Theta(2^n)$
* KMP algorithm: $\Theta(n)$

Overall cost: $\Theta(n+m)$
(assumed $|\Sigma|$ is a constant)

---

* Question: tell whether x contains a substring that mathes $(\# a b^* c + (ba)^*)$

mathes $(\#ab^*c + (ba)^*)$

(build a DFA for @ $(\#ab^*c + (ba)^*)$)

Context Free Languages (CFLs)

Consider $A = \{a^n b^n : n \geq 0\}$. A is an example of a CFL. It can be represented using a context-free grammar.

terminal

$$S \to \varepsilon \mid aSb$$

non-terminal

The grammar "generates" all $x \in A$:

$S \to \varepsilon$ we generated ε

$S \to aSb \to ab$ " " ab

$S \to aSb \to aaSbb \to aaaSbbb$

$\to aaabbb$

* Give a Context-Free Grammar (CFG) for the following language:

$A = \{x \in \{(,)\}^* :$ x is a "valid"

$A = \{ x \in \{ (,) \}^* : x$ is a "valid parenthesization$\}$

$( ) \in A$ ✓

$( )( ) \in A$ ✓

$)(\notin A$

$(( ) \notin A$