Claim 1

$\forall$ GNFA $G = (Q\ \Sigma\ \delta\ q_0\ q_f)$

with $k > 2$ states $\exists$

equivalent $G' = (Q', \Sigma, \delta', q_0, q_f)$

with $k-1$ states

pick state $k \neq q_0, q_f$

$Q' = Q - \{q_k\}$

δ'

$r_{ij} = \delta(q_i, q_j)$

$r_{ij}' = r_{ij} \cup r_{ik} r_{kk}^* r_{kj}$

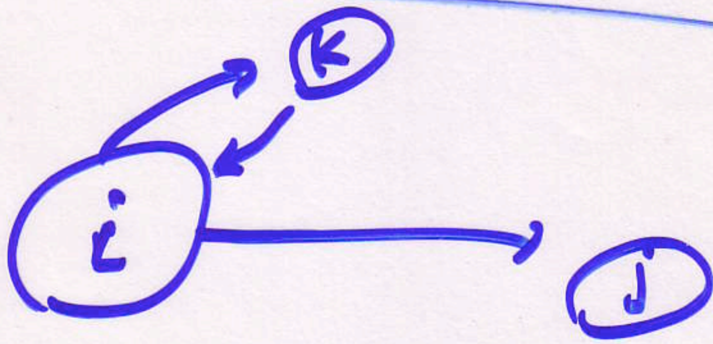
$\forall i, j \in Q'$

except $i \neq$ final
$j \neq$ initial

Claim 2

$L(r_{ij} \cup r_{ik} r_{kk}^{*} r_{kj})$

$= \{ x \mid$ could go from i to j reading x without passing through any intermediate state except possibly $k.$ $\}$



Claim 3

G & G' are equivalent

Claim 4 $\forall$ NFA $\exists$ equiv. reg. expr.

Proof: NFA $\to$ GNFA $\to$ 2-state GNFA $\to$ r.e.

by induction on k, using claim 1

Summary

L is regular $\Leftrightarrow$

$L = L(M)$ for some DFA M

$\Leftrightarrow$ $L = L(N)$ . . . . NFA N

$\Leftrightarrow$ $L = L(G)$ . . . . . GNFA G

$\Leftrightarrow$ $L = L(R)$ . . . . . reg. exp. R

Extended Regular Exp

compl((compl (r) ∪) · —)*)

~~Given~~ Write a program that, when given a reg. language L decides whether $L = \phi$.

how?

$$(\phi \cdot \phi \cup \phi) \cdot a$$

$$\text{Time} \geq \underbrace{2^{2^{2^{2}}}}_{} \updownarrow > \text{constant}$$

2

2^2

2^{2^2} 16

$2^{2^{2^2}}$ 65K

$2^{2^{2^{2^2}}}$) $2^{65000} \sim 10^{20000}$

Imagine a computer the size of a neutron ($\sim 10^{-15}$ meters diam.) capable of 10^{15} operations per sec

Radius of visible universe

$\sim 10^{10}$ light-years

$\times \pi \times 10^{7}$ sec/yr

$\times 3 \times 10^{8}$ m/s (speed of light)

$\approx 10^{26}$ meters

Volume $\sim (10^{26})^3 = 10^{78}$ m^3

So packing visible universe with these neutron-size computers gives $\sim \frac{10^{78}}{(10^{-15})^3} = 10^{123}$ processors

@ 10^{15} ops/sec $\times \pi \times 10^{7}$ sec/yr $\times 10^{10}$ yr $\times 10^{123}$ processors $\approx 10^{155}$ ops since the dawn of time ...