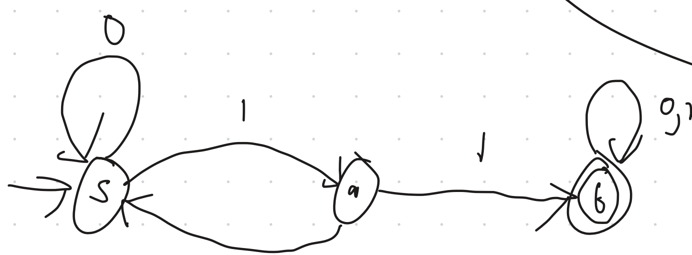
Announcements:

Homework parties: Siebel 0216

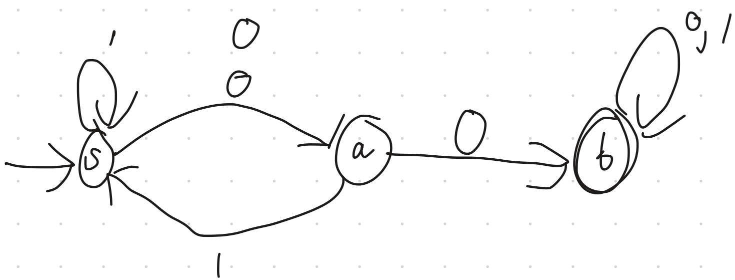
Thur 6-8 (today!)

Sun 4-6

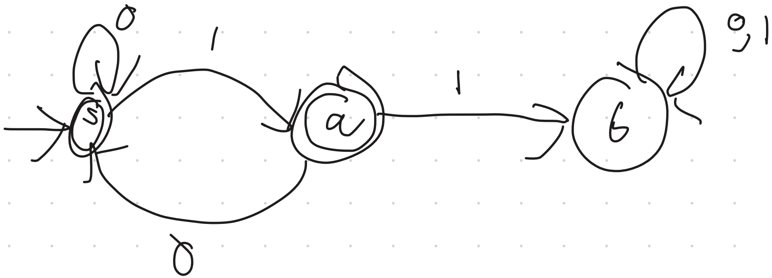
Mon 6-8



11 as a substring M_{11}



00 as a substring M_{00}



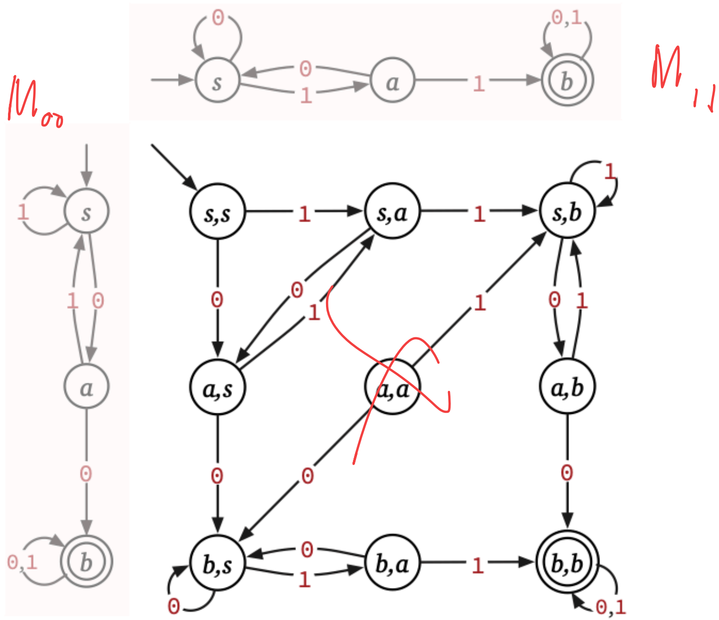
does not have 11 as a substring $\overline{M_{11}}$

Product construction: states (p, q) p from M_{00}, q from M_{11}

start at (s, s') s starts M_{00}, s' starts M_{11}

$(p,q) \xrightarrow{a} (p', q')$ if $p \xrightarrow{a} p'$, $q \xrightarrow{a} q'$

accept (p, q) where M_{00} accepts p, M_{11} accepts q



Building a DFA for the language of strings containing both 00 and 11.

DFA over alphabet Σ:

Q: set of state

$s \in Q$: start state

$A \subseteq Q$: accept states

$\delta : Q \times \Sigma \to Q$ transition function

extend transition function $\delta^* : Q \times \Sigma^* \to Q$

$$\delta^*(q, w) := \begin{cases} q & w = \epsilon \\ \delta^*(\delta(q, a), x) & w = ax \end{cases}$$

Given two DFAs $M_1 = (Q_1, s_1, A_1, \delta_1)$

$M_2 = (Q_2, s_2, A_2, \delta_2)$

$L(M) = \{w \mid M \text{ accepts } w\} = \{w \mid \delta^*(s, w) \in A\}$

Can we build a DFA for $L(M_1) \cap L(M_2)$?

$M = (Q, s, A, \delta)$

$Q = Q_1 \times Q_2 = \{(p, q) \mid p \in Q_1 \text{ and } q \in Q_2\}$

$s = (s_1, s_2)$

$A = \{(p, q) \mid p \in A_1 \ \underline{\text{and}} \ q \in A_2\}$

$\delta((p, q), a) = (\delta_1(p, a), \delta_2(q, a))$

Thm: $L(M) = L(M_1) \cap L(M_2)$

Lemma: $\delta^*((p,q),w) = (\delta_1^*(p,w), \delta_2^*(q,w))$ for $p \in Q_1$, $q \in Q_2$, & $w \in \Sigma^*$

Proof: Let $p \in Q_1$, $q \in Q_2$, and $w \in \Sigma^*$ be arbitrary.
Assume for all strings x such that $|x| < |w|$ and for any states $p' \in Q_1$ and $q' \in Q_2$
$\delta^*((p',q'),x) = (\delta_1^*(p',x), \delta_2^*(q',x))$.

There are two cases.

Suppose $w = \epsilon$.

$$
\begin{aligned}
\delta^*((p,q),w) &= \delta^*((p,q),\varepsilon) && w=\varepsilon \\
&= (p,q) && \text{def. } \delta^* \\
&= (\delta_1^*(p,\varepsilon), \delta^*(q,\varepsilon)) && \text{def. } \delta_1^*, \delta_2^* \\
&= (\delta_1^*(p,w), \delta_2^*(q,w)) && w=\varepsilon
\end{aligned}
$$

Suppose $w = ax$.

$$
\begin{aligned}
\delta^*((p,q),w) &= \delta^*((p,q),ax) && w=ax \\
&= \delta^*(\delta((p,q),a),x) && \text{def. } \delta^* \\
&= \delta^*((\delta_1(p,a), \delta_2(q,a)),x) && \text{def. } \delta \\
&= (\delta_1^*(\delta_1(p,a),x), \delta_2^*(\delta_2(q,a),x)) && \text{IH} \\
&= (\delta_1^*(p,ax), \delta_2^*(q,ax)) && \text{def. } \delta_1^*, \delta_2^*
\end{aligned}
$$

$= (\delta_1^*(p, w), \delta_2^*(q, w))$ $\quad w = ax$

In all cases, $\delta^*((p,q), w) = (\delta_1^*(p,w), \delta_2^*(q,w))$ $\square$

Call a language L <u>automatic</u> if L is accepted by some DFA.

They are <u>closed</u> under simple boolean opperations.

Thm: Let L_1 and L_2 be automatic languages, the following are automatic: regular

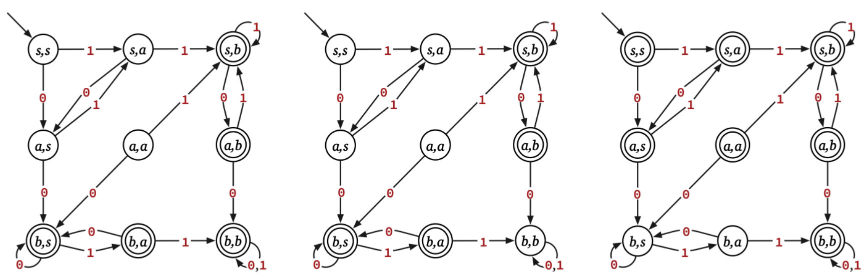
$\overline{L_1} = \Sigma^* \setminus L_1$ $\qquad A = Q_1 \setminus A_1$

$L_1 \cup L_2$ $\qquad A = \{(p,q) \mid p \in A_1 \ \underline{\text{or}}\ q \in A_2\}$

$L_1 \cap L_2$ $\qquad$ Same product constructions except...

$L_1 \setminus L_2$ $\qquad$ " $\quad p \in A_1\ \underline{\text{but}}\ q \notin A_2$

$L_1 \oplus L_2$ $\qquad$ " $\quad p \in A_1\ \underline{\text{xor}}\ q \in A_2$



(a) $\qquad$ (b) $\qquad$ (c)

00 or 11 $\qquad$ 00 or 11 but not both $\qquad$ 00 ⇒ 11

Thm (Kleene): The automatic languages are precisely the regular languages

Given a regular expression R, there is a DFA s.t. $L(M) = L(R)$

Given DFA M, there is a regular expression R such that $L(R) = L(M)$

Proof uses nondeterministic finite-state automata (NFAs)

Thm: If L_1 and L_2 are automatic, so are $L_1 \circ L_2$ and L_1^*

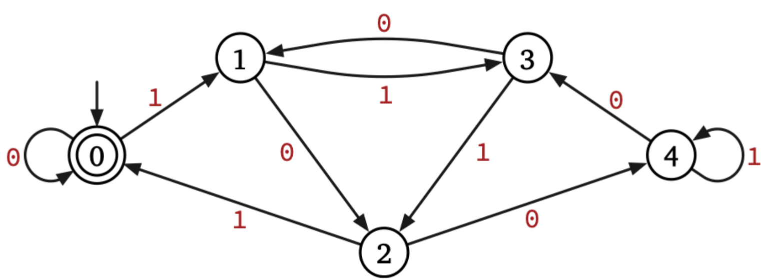
DFAs have no memory. You don't know how you got to a particular state & it cannot matter.

Good: Given two states that always agree on rejection or acceptance given any string, we combine them.

Bad: Given two strings x & y and a language L.

If there is a string z such that exactly one of xz or yz are in L, then any DFA for L must send x and y to distinct states.

z is a <u>distinguishing suffix</u> of x and y with respect to L.



binary numerals divisible by 5

$x = 01$

$y = 0011$

distinguished by $z = 01$

$$\overset{x}{0}1\overset{z}{0}1 \in L$$

$$0\underset{y}{0}11\underset{z}{0}1 \notin L$$