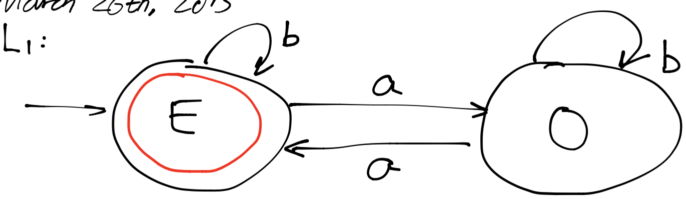
Lecture 20

March 26th, 2015

L_1:



Recognizes/Accepts language L_1 over $\Sigma = \{a, b\}$ of strings where # of as is odd.

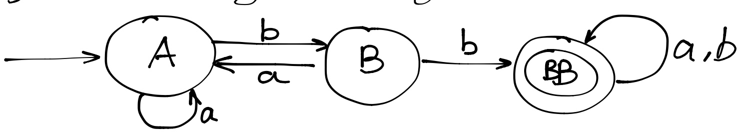
Let L_2 be: Language over Σ ..... # of a's is even.

complement of L_1

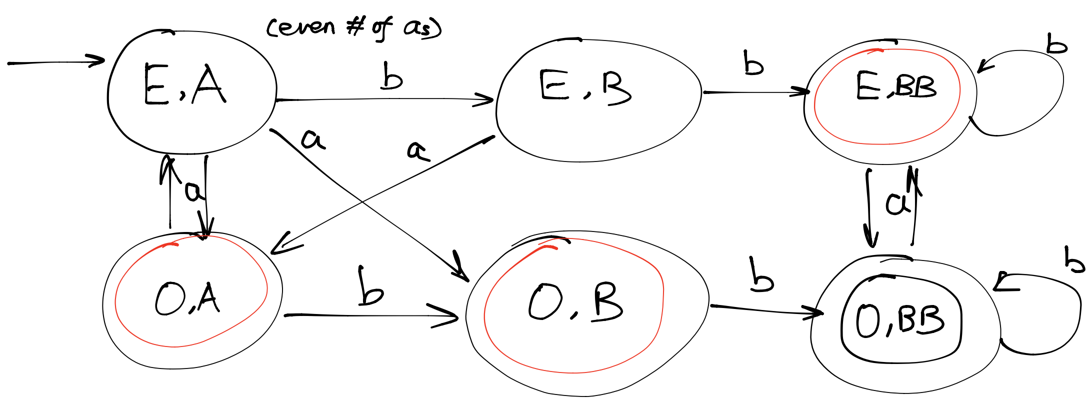
All strings over Σ that are not in L_1

Can make a machine for negation of the property P where for each string s over Σ, P(s) is: S contains odd # of as.

L_3 over Σ: strings containing bb somewhere



Make $L_4 = L_1 \cap L_3$: odd # of as, and bb somewhere



$L \circ L'$ is the "concatenation" of L and L'

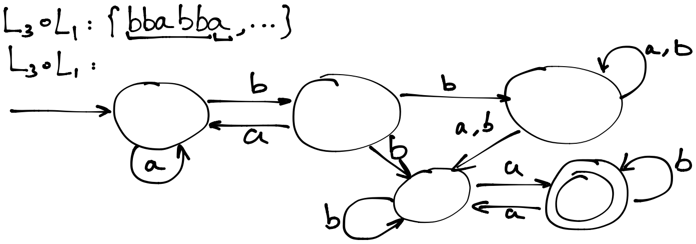
Defined as $\{ss' : s \in L, s' \in L'\}$

(concatenate)

$L_1 \circ L_3$: {a bb, a abb, a aa bbaabb, bbabb, ...}

$L_3 \circ L_1$: {bba bba, ...}

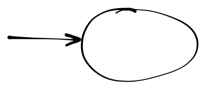
$L_3 \circ L_1$:



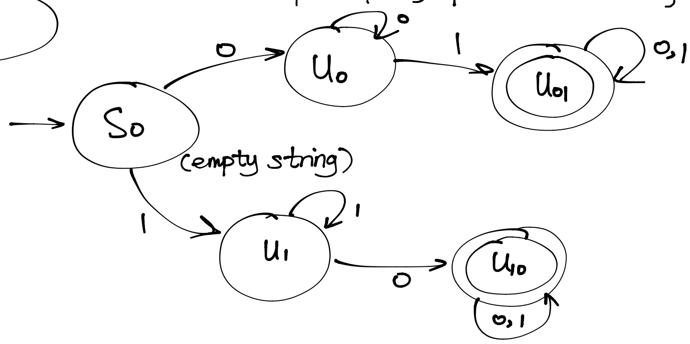
Tutorial

$b \in S$ means S contains at least 1 occurence of b

empty string



$L := \{S \in \{0,1\}^* \mid 0 \in S \text{ and } 1 \in S\}$



Σ	$1 \notin L$
0	$1 \in L$

0	$1 \in L$
1	$1 \notin L$

Σ	$0 \notin L$
1	$0 \in L$

01	$0 \in L$
0	$0 \notin L$

01	$1 \in L$
1	$1 \notin L$

01	$1 \in L$
Σ	$1 \notin L$

In any DFA that computes L, the string 01 must land in a state distinct from the state that $0, 1, \varepsilon$ land in.

Defn: For M a DFA and $S \in \{0,1\}^*$, Let $dest(s)$ be the state that M is in M after reading S.

Defn: $\mathcal{L}(M)$ is the language computed by M.

$\forall M,\ \mathcal{L}(M) = L,\ dest(\varepsilon) \neq dest(0)$ ①

and $\underbrace{(dest(\varepsilon) \neq dest(1) \neq dest(0)}_{\text{all distinct}})$

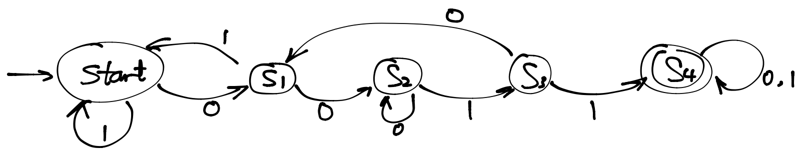
$dest_M(0) \in \{dest_M(1), dest_M(\varepsilon)\}$ ②

and $dest_M(0\ 1) \notin \{dest_M(0), dest_M(1), dest_M(\varepsilon)\}$ ③

①②③ $\Rightarrow |\{det_M(\varepsilon), det_M(0), det_M(1), det_M(01)\}| = 4$

L a language, $S_1, \cdots, S_n \in \{0,1\}^*$, and for every pair $S_i \neq S_j$, there is a suffix $t \in \{0,1\}$ s.t $S_i t \in L$ and $S_j t \notin L$ or vice-versa

Then any DFA computing L must have n states.



1. Find the shortest path : 0011 in language

$00\cdots011 \in L$

$11\cdots00\cdots11 \in L$

$1^*000^*111^* \in L$

$\underbrace{1^*0^*}0011\underbrace{1^*0^*} \subseteq L$

? could be any string

$S \in \{0,1\}^*$ reaches S_4 means that at least 0011 in S.

S_4: string is in L

S_3: last three characters are 001

S_2: last 2 characters are 00, and no prefix is in L.

S_1: last character is 0, and $\neg S_2$ and $\neg S_4$

Start: Σ or last character is 1, and not preceeded by 00.