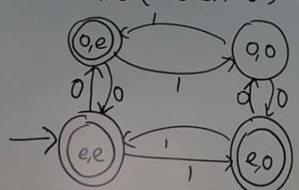
CS 228 Tutorial 3

1a. Strings over $\{0,1\}^*$

$\#_0 \cdot \#_1$ is even.

i.e. either even 0's or even 1's



NOT FO-definable.

EF games

For every r

need x_r, y_r $x_r \in L$, $y_r \notin L$

Game of r-moves played on (x_r, y_r)

has winning strategy for Duplicator

$0^{2r}1$ $\quad$ $0^{2r+1}1$

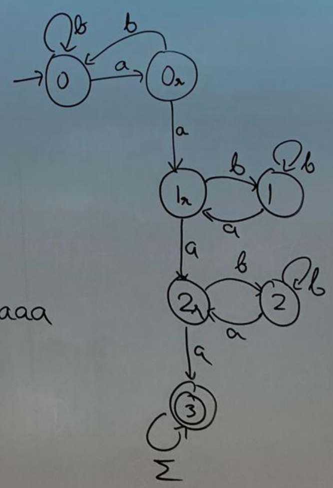
x_r $\quad$ y_r

CS 228 Tutorial 3

1b. Three a's that are immediately followed by an 'a'

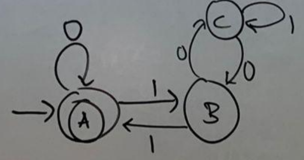
$$\exists x_1, x_2, x_3, y_1, y_2, y_3 . (x_1 < x_2 < x_3) \wedge \bigwedge_{i=1}^{3} S(x_i, y_i)$$

$$\wedge \bigwedge_{i=1}^{3} Q_a(x_i) \wedge Q_a(y_i)$$



aaa

1c $(0+1(01^*0)^*1)^*$



- Regex to NFA
- Commonsense NFA to DFA
- DFA minimisation
  - i.e. collapse "equivalent" or redundant states

	A	B	C
ε	A	B	C
0	A	C	B
1	B	A	C
01	B	C	A
10	C	A	B
010	C	B	A

NOT FO-definable.

all 6 permutations are possible

$\sigma^k = \sigma^{k+1}$ is not going to hold $\forall \sigma$

CS 228 Tutorial 3

Trick (USE ONLY IN ROUGH WORK)

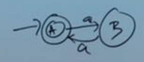
- Mathematically sound
- But not taught in this course
- If you get blank in EF games, use this to get answer, then think of proof via EF.

Consider $\Sigma = \{a\}$

$L = (aa)^*$

FO define ≡ Recognised by aperiodic monoid

1) Draw minimal DFA



2) Recall $\hat{\delta}$

$\hat{\delta}: Q \times \Sigma^* \to Q$

Define related $\tau(w)$

$\tau(w): Q \to Q$

$\tau(w)(q) = \hat{\delta}(q, w)$

Define

$T = \{\tau(w) \mid w \in \Sigma^*\}$

FINITE. Because Q finite

	A	B
ε	A	B
a	B	A

Enumerate the elements of T

Multiplication operation on T

$\sigma_1 \sigma_2(q) = \sigma_2(\sigma_1(q))$...

T is function composition.

Recall $\sigma_1, \sigma_2 : Q \to Q$

Interested in τ^k

FO definable iff $\exists k$ s.t. $\forall \tau \in T$ $\tau^k = \tau^{k+1}$

CS 228 Tutorial 3

2. L_1, L_2 : FO-definable.

Is $L_1 \cdot L_2$ FO-definable.

$\{\}$: $\exists x . x \neq x$

$\{\varepsilon\}$: $\forall x . x \neq x$

$L_1' \equiv L_1 \setminus \{\varepsilon\}$: $\varphi_{L_1} \wedge \exists x . x = x$

If $\varepsilon \in L_1$

$(\varphi_1 \wedge \exists x . x = x) \vee \forall x (x \neq x)$

If either L_1 or L_2 are $\{\}$ then $L_1 \cdot L_2 = \{\}$

ε in	
none	$L_1 \cdot L_2$
L_1	$L_1' \cdot L_2 + L_2$
L_2	$L_1 \cdot L_2' + L_1$
both	$L_1' \cdot L_2' + L_1 + L_2$

WLOG assume L_1, L_2 are non-empty and do not contain ε.

CS 228 Tutorial 3

2.

Assume $\varphi_{L_1}, \varphi_{L_2}$ use only $\exists$ quantifier
- Use disjoint sets of variables

$$\varphi_{L_1 \cdot L_2} \equiv \exists x_{fin}, y_{start}\ S(x_{fin}, y_{start}) \land \varphi'_{L_1} \land \varphi'_{L_2}$$

φ'_{L_1} is the same as φ_{L_1} EXCEPT

$\exists x \mapsto \exists x.\ x < x_{fin} \land \ldots$

φ'_{L_2} is same as φ_{L_2} except

$\exists y \mapsto \exists y.\ y_{start} \leq y \land \ldots$

x_{fin}, y_{start} are FRESH variables here.

WLOG assume L_1, L_2 are non-empty and do not contain ε.

CS 228 Tutorial 3

3 $h: \Sigma^* \to \Gamma^*$

$h(\varepsilon) = \varepsilon$

$h(a) \in \Gamma^*$

$\forall w_1, w_2 \in \Sigma^*\quad h(w_1) \cdot h(w_2) = h(w_1 w_2)$

$L \subseteq \Sigma^*$

$h(L) = \{h(w) \mid w \in L\}$

a) L is regular

So let $A: (Q, \Sigma, q_0, \delta, F)$ be DFA accepting it.

Can we construct NFA A' accepting $h(L)$

- Recall then $\uparrow S,\ S \subseteq Q$ we defined last time.
- Will help if $h(a) = \varepsilon$.

Idea

Same set of states. Replace the label a on transitions by $h(a)$

What if $|h(a)| > 1$?

It is still finite

$\delta(q, a) = q'$ $\qquad a_1 a_2 \ldots a_n$

$q \xrightarrow{a} q'$

$q \xrightarrow{a_1} \circ \xrightarrow{a_2} \circ \cdots \circ \xrightarrow{a_n} q'$

3 $h: \Sigma^* \to \Gamma^*$

$h(\varepsilon) = \varepsilon$

$h(a) \in \Gamma^*$

$\forall w_1, w_2 \in \Sigma^*\quad h(w_1) \cdot h(w_2) = h(w_1 w_2)$

$L \subseteq \Sigma^*$

$h(L) = \{h(w) \mid w \in L\}$

b) $\Sigma = \{a\}$ $\quad \Gamma = \{b\}$

$L = \Sigma^*$ $\quad h(a) = bb$

then $h(L) = (bb)^*$

not FO-definable

as seen earlier in this tutorial

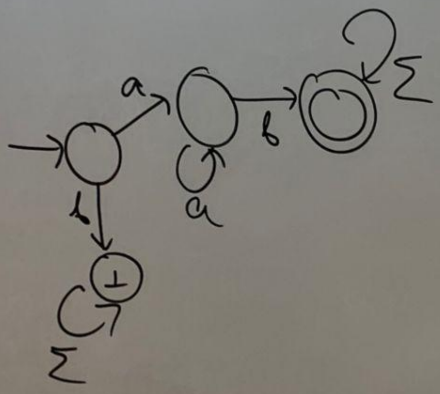
T

4 $\Sigma = \{a, b\}$

Sat : ab

not valid: b, aa

$a^* a b \Sigma^*$



4 $\Sigma = \{a, b\}$, axioms: exactly one letter in a position

Whenever I see an 'a' at position x

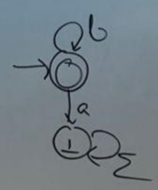
- I will see a 'b' later at y.
- There are two distinct later positions z_1, z_2
  - Both these can't be 'b'
    - i.e. at least one of them is an 'a'

Key: No finite word that contains an "a" can satisfy formula.

sat: b^*

not valid

a



CS 228 Tutorial 3

5\. I have a set of trees, each tree is itself a set of leaves.

How to convert to DFA

States

2^{2^Q}

Eg. $\{\{q_1\}, \{q_2, q_3\}\}$

denote trees

denote leaves

$q \in Q_\exists$ then computation takes non-deterministic choice } make new trees

q $\quad q_1 \vee q_2$

$q \to q_1 \quad q \to q_2$

$q \in Q_\forall$ computation splits into threads, each of which are expected to be accepting } Split tree into new branches

q $\quad q_1 \wedge q_2$

$q \to q_1, q_2$

Accept: exists a tree, all leaves accepting.

Complement?

Flipping $Q_\exists$ and $Q_\forall$

Flipping accept states

6(a) $\underbrace{(ab)^n b (ab)^n}$ $\quad (ab)^m (ab)^n$

contains bb as substring

abbab (S_1, S_2)

ababab (D_1, D_2, S_3)

D_3: oops.

$$\exists x_1 x_2 \;\; x_1 < x_2 \;\; Q_b(x_1) \wedge Q_b(x_2) \wedge \neg \exists y.\; x_1 < y < x_2$$

6(c) Very similar to (a)

Just more cumbersome.

6 (b)

$(aa)^n b (aaa)^n$ $(aaa)^n b (aa)^n$

$2n$ or $2n+1$, forgive my off-by 1

Spoiler picks $(aa)^n b$

and then demonstrates

Can't pick another a to left of

b in first word.

If want to prove L

not FO-definable

for all n

pick $w_n \in L$, $x_n \notin L$,

Show how Duplicator wins

on (w_n, x_n)