Lecture 7 - May 27

Lexical Analysis

DFA: Formulation

DFA δ : Non-Recursive vs. Recursive

NFA: Motivation

Announcements/Reminders

- Assignment 1 to be released next Monday
- Review Slides on Math posted

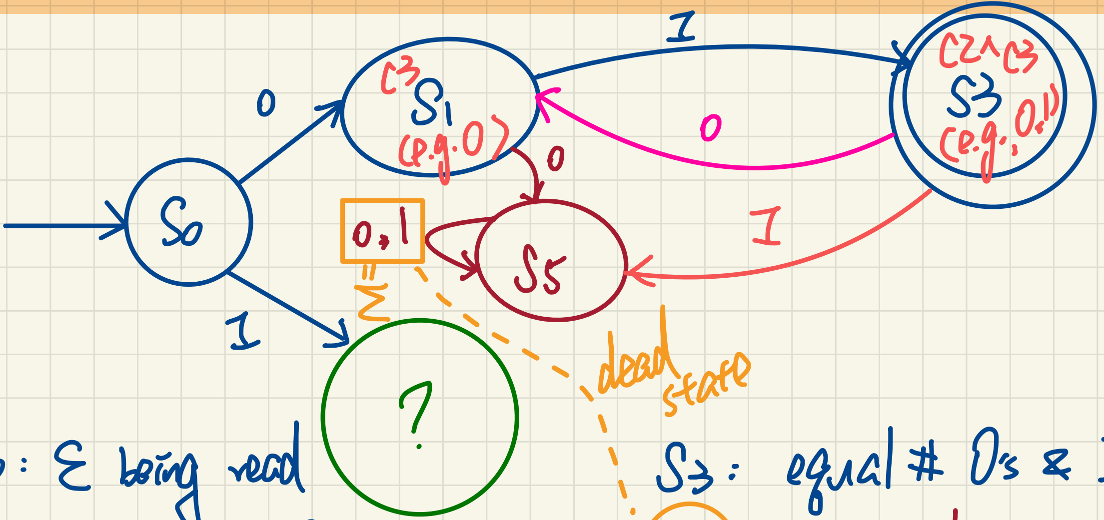
DFA: Exercise

$\#_{\text{outgoing}}$ tran $= |\Sigma| = |\{0,1\}| = 2$

Draw the **transition diagram** of a **DFA** which **accepts/recognizes** the following language:

c1 c2 c3

{ w | w ≠ ε ∧ w has equal # of alternating 0's and 1's }



S0 : ε being read

S1 : read more 0's than 1's

S3 : equal # 0's & 1's ∧ $(01)^+$

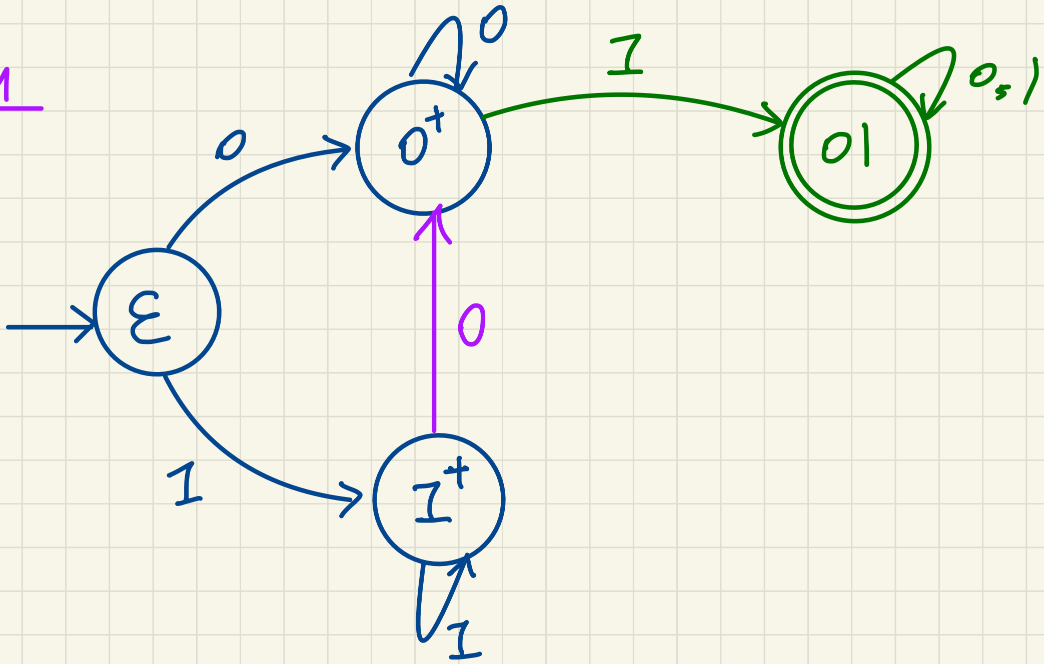
S5 : not alternating

DFA Exercise

alt: ends with 01.

$L = \{ w \mid w \in \{0,1\}^* \wedge w \text{ contains } 01 \text{ as a substring}\}$

William



set of strings recognized/accepted by DFA

slide 24

DFA: Formulation (1) M

$\delta(q_0, a_1) = q_1,\ \delta(q_1, a_2) = q_2$

A *deterministic finite automata (DFA)* is a 5-tuple

model $M = (Q, \Sigma, \delta, q_0, F)$ accept states

states · alphabet · initial/start state

Language of a DFA

w s.t. |w| = n

$$L(M) = \{ a_1 a_2 \ldots a_n \mid 1 \le i \le n \land \delta(q_{i-1}, a_i) = q_i \land q_n \in F \}$$

$\delta(q_i, a_{i+1}) = q_{i+1}$

e.g., 0101

$a_1\ a_2\ a_3\ a_4$

$|a_1|a_2|a_3|a_4|$

$q_0\ \ q_1\ \ q_2\ \ q_3\ \ q_4$

$\delta(s_0, \underset{a_1}{0}) = s_1$

$\delta(s_1, \underset{a_2}{1}) = s_3$

$\delta(s_3, \underset{a_3}{0}) = s_1$

$\delta(s_1, \underset{a_4}{1}) = s_3$

accepted: $s_3 \in F = \{s_3, s_4\}$

(transition function)

$\delta : (Q \times \Sigma) \to Q$

source state $q \in Q$, $c \in \Sigma$

given an input pair (q, c)

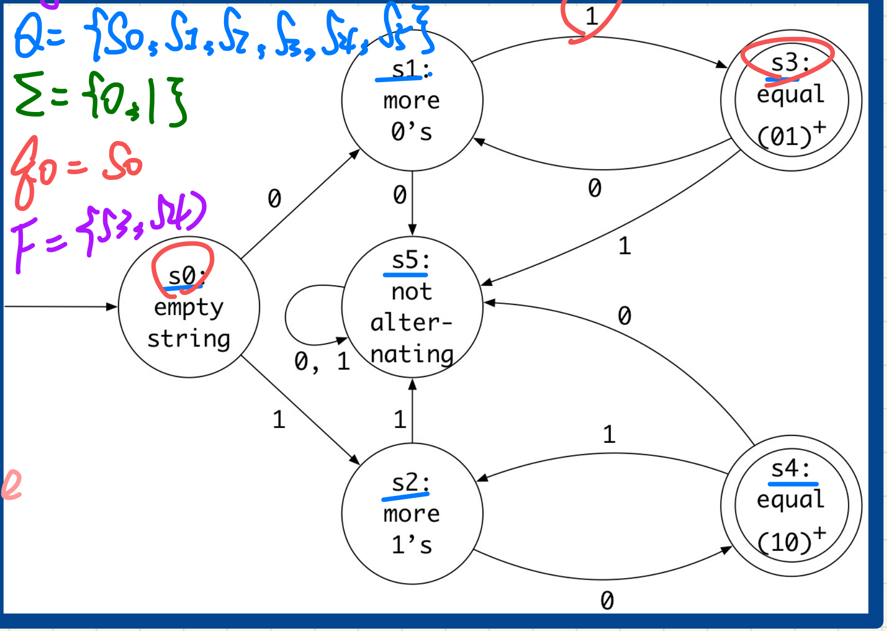
always return a single state $q' \in Q$ destination state

$Q = \{s_0, s_1, s_2, s_3, s_4, s_5\}$

$\Sigma = \{0, 1\}$

$q_0 = s_0$

$F = \{s_3, s_4\}$



$$\delta : (Q \times \Sigma) \boxed{\longrightarrow} Q$$

total function.

study question for Math Review lecture: Is this alt. formulation appropriate for DFA?

Alternative formulation of δ:

$$\delta : (Q \times \Sigma) \boxed{\not\longrightarrow} Q$$

partial function.

$* \; w = \boxed{x}\,a \to$ last char.

prefix

DFA: Formulation (2)

A **deterministic finite automata (DFA)** is a 5-tuple

$$M = (Q, \Sigma, \delta, q_0, F)$$

Language of a DFA

$\delta : (Q \times \Sigma) \to Q$

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

We may define $\hat{\delta}$ recursively, using δ!

$$\hat{\delta}(q, \epsilon)^{*} = q$$

$$\hat{\delta}(q, \boxed{xa}) = \delta(\hat{\delta}(q, x), a)$$

where $q \in Q$, $x \in \Sigma^*$, and $a \in \Sigma$

e.g., $\overset{x}{\boxed{01}}\,\overset{a}{\boxed{0}}$

$w = \overset{x}{\boxed{01}}\,\overset{a}{0}$

$x = 01$

$a = 0$

$q \xrightarrow{xa} ?$

$q \xrightarrow{x} q' \xrightarrow{a} ?$

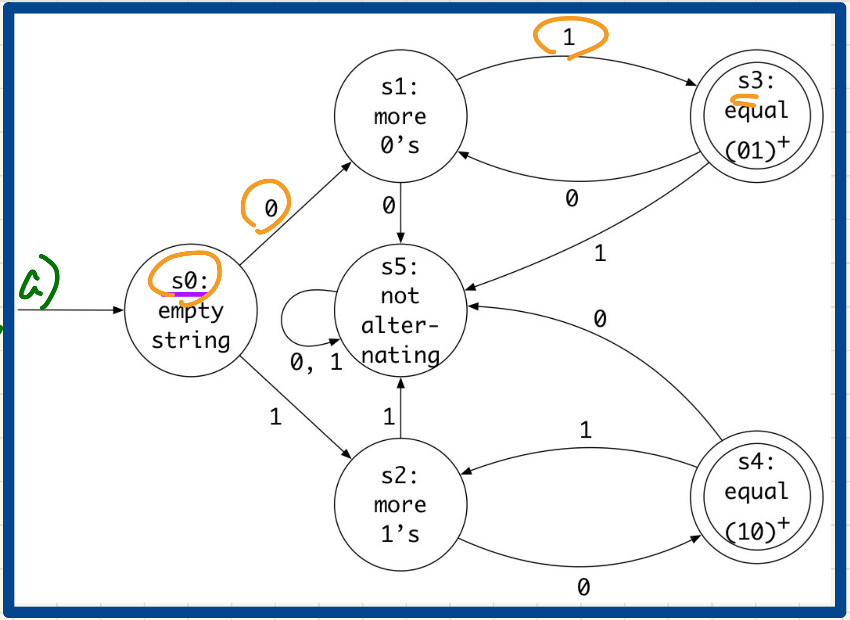
$q' = \hat{\delta}(q, x)$

$? = \delta(q', a)$

$\hat{\delta}(s_0, \overset{x}{\boxed{01}}\overset{a}{\boxed{0}})$

$= \delta(\overset{s_3}{\boxed{\hat{\delta}(s_0, 01)}}, 0) = s_1$

$$L(M) = \{ w \mid w \in \Sigma^* \wedge \hat{\delta}(q_0, w) \in F \}$$



DFA vs. NFA

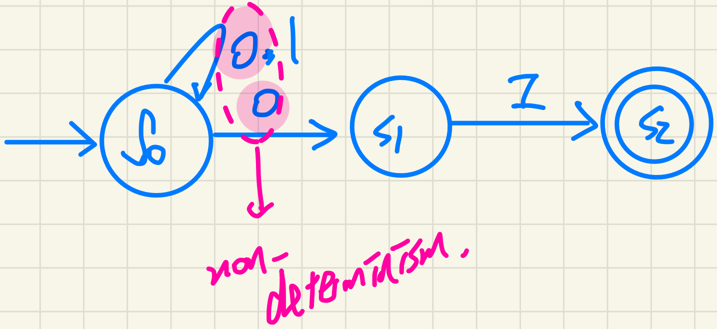
Problem: Design a DFA that accepts the following language:

$$L = \{ \ x01 \mid x \in \{0,1\}^* \ \}$$

That is, L is the set of strings of 0s and 1s ending with 01.

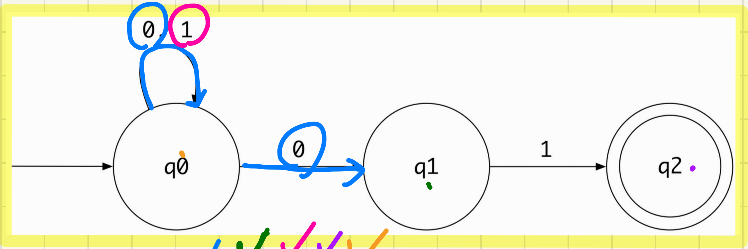
Exercise:
DFA

A *non-deterministic finite automata (NFA)* that accepts the same language:



NFA Behaviour ≈ Alternative Universe

Obviously the **time continuum** has been disrupted, creating this new temporal event sequence resulting in this **alternate reality**. [redacted]



Trace: 00101

Overall, 00101 is accepted.