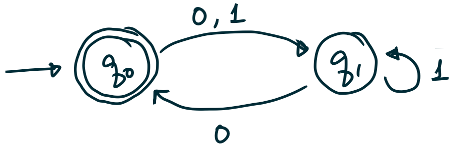
MATH 2301

* Deterministic finite automata (DFA)



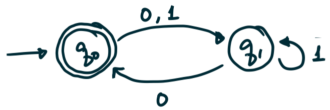
← state diagram of a DFA

Def: A DFA consists of the following pieces of data:

1) An alphabet Σ
2) A set Q of "states"
3) A start state $q_0 \in Q$ ← a hanging incoming arrow
4) A set of accept states $A \subseteq Q$ ← doubly-circled
5) A transition function $\delta : Q \times \Sigma \to Q$

state you're at — letter you read — where you end up

** Example



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

$Q = \{q_0, q_1\}$

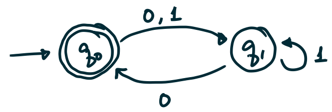
q_0 the start state

$A = \{q_0\}$

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

Q	Σ	output in Q
q_0	0	q_1
q_0	1	q_1
q_1	0	q_0
q_1	1	q_1

** Reading strings (examples)



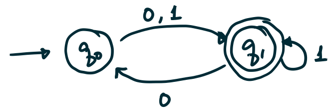
Let $\boxed{w = 01101}$

1) Start at the start state q_0
2) Read w from left to right, letter by letter, and follow the labelled arrows.

State	Letter read
q_0	0
q_1	1
q_1	1
q_1	0
q_0	1
q_1	(end)

Since we ended at q_1 and $q_1 \notin A$, we REJECT.

Other accepted words include
$w = \varepsilon$
$w = 00$, $w = 010$



← This machine will accept $w = 01101$

It will accept $w = 0$, $w = 111$, $w = 001$

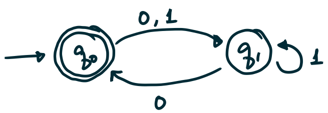
It will reject $w = 10$, $w = \varepsilon$, $w = 00$

** The language of a DFA

Let M be a DFA. The language of M, denoted $L(M)$, is the set of strings that M accepts.

** Question: Is there any relationship between languages of regular expressions and languages of DFAs?

** Example

$M =$ 

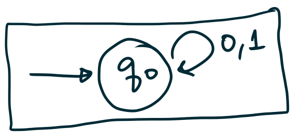
$L(M) = L(r)$

where $r = ((0|1)1^*0)^*$

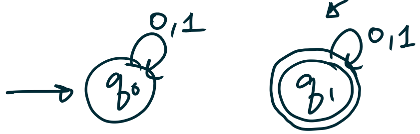
** Let's try to "convert" regexs into machines, ie, given a regex r, we'll try to build M such that $L(r) = L(M)$.

"Easy" cases, say $\Sigma = \{0,1\}$

1) $r = \phi$, $L(r) = \phi$

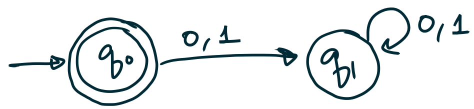


⤳ this machine rejects all strings, yay.



(and many other options ...)

2) $r = \varepsilon$, $L(r) = \{\varepsilon\}$



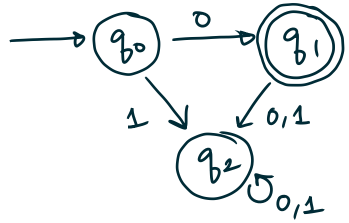
⤳ only accepts ε.

(and many other options ...)

3) $r = a$ for some $a \in \Sigma$

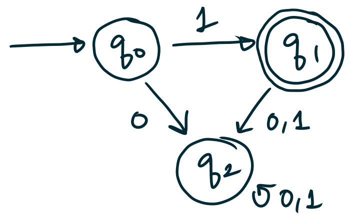
$L(r) = \{a\}$

E.g. $r = 0$



⤳ only accepts 0

E.g. $r = 1$



⤳ only accepts 1

4) $r = r_1 r_2$, $L(r) = L(r_1) \circ L(r_2)$

Q: Given M_1 & M_2 DFAs such that $L(M_i) = L(r_i)$, construct a machine M, such that $L(M) = L(r)$