HW1 due 9pm.
GPS & HW due next Mon + Tue.

Language: set of strings.

Regular languages: $\emptyset$, a single string $\{w\}$, union, concatenation, or kleene closure of other regular languages.

Regular expressions: $\emptyset$, w, $R_1 + R_2$, $R_1 R_2$, or R^*

Ex: find a regular expression for non-negative binary numerals divisible by 4 with no redundant leading 0s.

~~00100~~ 100 ✓ 1~~0~~1

$0 + 1(0+1)^*00$

Want to take a binary string $w \in \{0,1\}^*$ and decide if # 1s is divisible by 3.

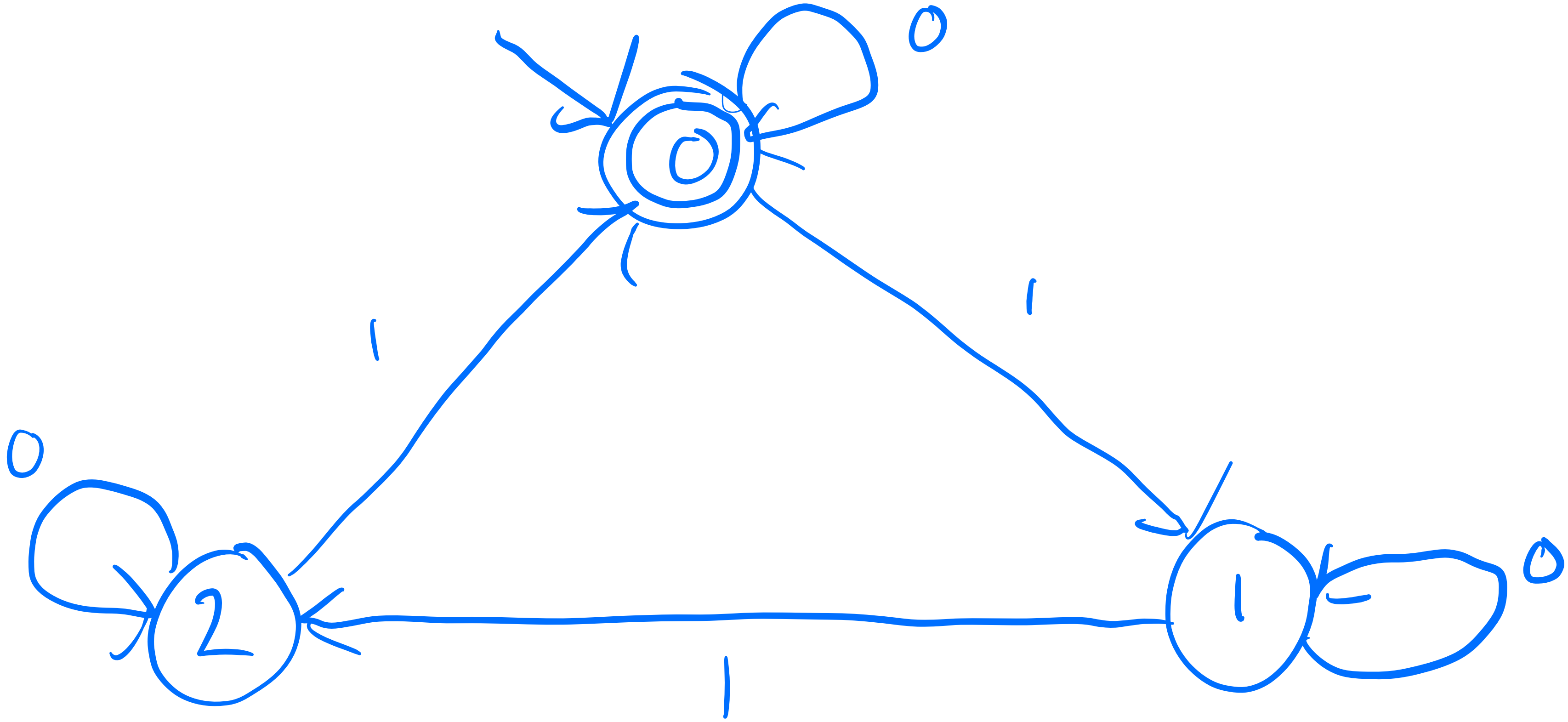
(i.e. $\#(1, w) \equiv 0 \pmod 3$)

Want to look at symbols one-by-one.

```
Num1sDivisbleBy3(w[1..n])
  rem ← 0
   for i ← 1 to n
     if w[i] = 1
       rem ← rem + 1 (mod 3)
  return (rem = 0)
```

start

state/symbol	0	1	return true/false if final state
0	0	1	yes
1	1	2	no
2	2	0	no



finite-state machine or

deterministic finite-state automata

(DFA)

A DFA consists of five components:

- an arbitrary finite set Σ, the alphabet
- an arbitrary finite set Q, the states
- an arbitrary transition function

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

- a start state $s \in Q$
- a subset $A \subseteq Q$, accept states

Given an input string $w \in \Sigma^*$, DFA reads symbols from left to right.

If at state q + read a, go to state

$$\delta(q, a).$$

After reading w, it <u>accepts</u> w if in a state in A & <u>rejects</u> w otherwise.

<u>extended transition function:</u>

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

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

(where you end up after reading all of w)

DFA <u>accepts</u> iff $\delta^*(s, w) \in A$.

Original example:

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

$$Q = \{0, 1, 2\}$$

$$\delta(q, a) = \begin{cases} q & \text{if } a = 0 \\ q+1 \pmod{3} & \text{if } a = 1 \end{cases}$$

$$s = 0$$

$$A = \{0\}$$

$$w = 01010110101$$

$$\begin{aligned}
\delta^*(s, w) &= \delta^*(0, 01010110101) \\
&= \delta^*(\delta(0,0), 1010110101) \\
&= \delta^*(0, 1010110101) \\
&= \delta^*(\delta(0,1), 010110101) \\
&= \delta^*(1, 010110101) \\
&\vdots
\end{aligned}$$

$= f^{*}(0, \epsilon)$

$= 0 \in A$

accept!

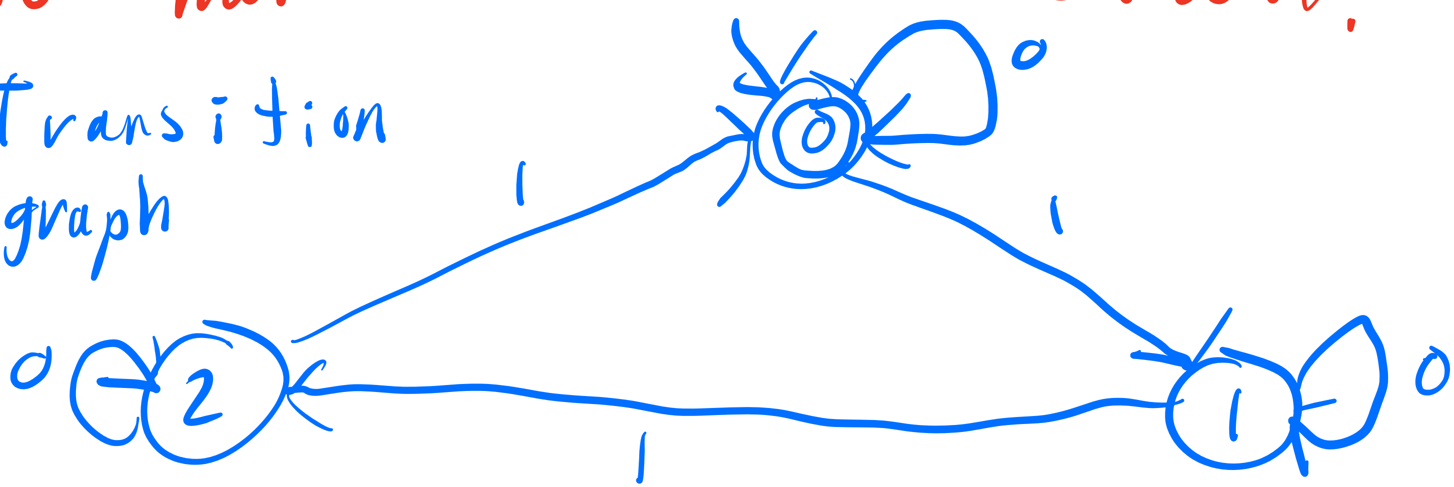
Describing a DFA:

1) Table

q	$\delta(q,0)$	$\delta(q,1)$	$q \in A$
0	0	1	True
1	1	2	False
2	2	0	False

Also name the start state!

2) Transition graph

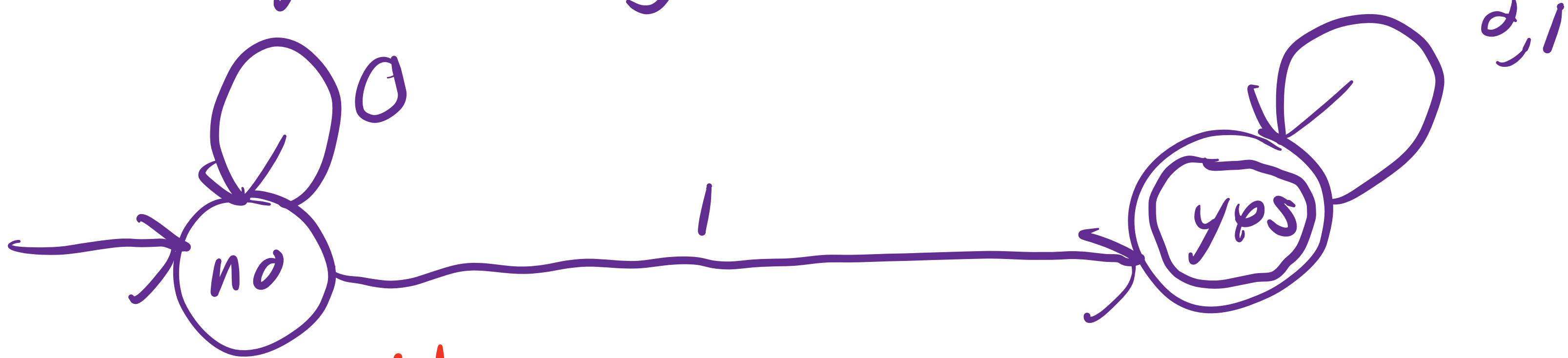


Don't forget start state arrow $\to\bigcirc$
or accept state circles $\circledcirc$.

3) math notation

$$\delta(\_\,,\_) = \begin{cases} & \text{if} \\ & \text{if} \\ & \vdots \end{cases}$$

Ex: Binary strings contain a 1.



Ex: all binary!! numerals w that are a multiple of 5.

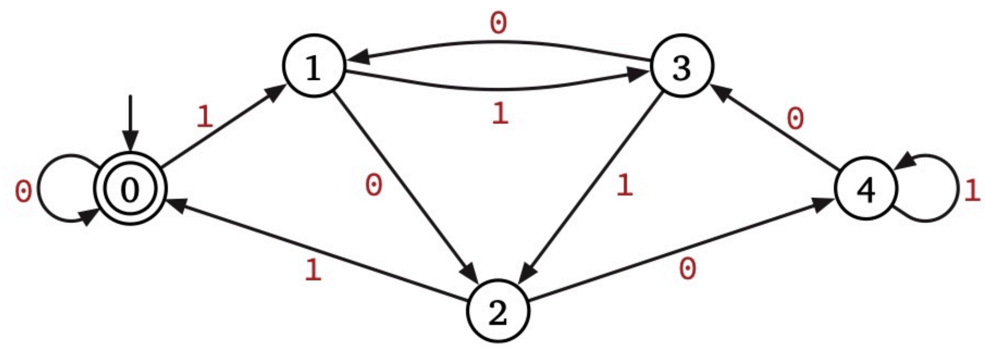
0110<u>0</u>

<u>MULTIPLEOF5($w[1..n]$):</u>

$rem \leftarrow 0$
for $i \leftarrow 1$ to n
    $rem \leftarrow (2 \cdot rem + w[i]) \bmod 5$
if $rem = 0$
    return TRUE
else
    return FALSE

q	$\delta[q,0]$	$\delta[q,1]$	$A[q]$
0	0	1	TRUE
1	2	3	FALSE
2	4	0	FALSE
3	1	2	FALSE
4	3	4	FALSE

or $\delta(q,a) = (2q + a) \bmod 5$



start at $s=0$.

Ex: begin with 001.