GPS 1 due 9pm today

HW 1 due 9pm tomorrow

homework policies on website

cite all sources or say none

transcripts with CLMs

Standard rubrics on wpt site

All strings $w \in \{0,1\}^*$ s.t. $\#(1,w) \equiv 0 \mod 3$?

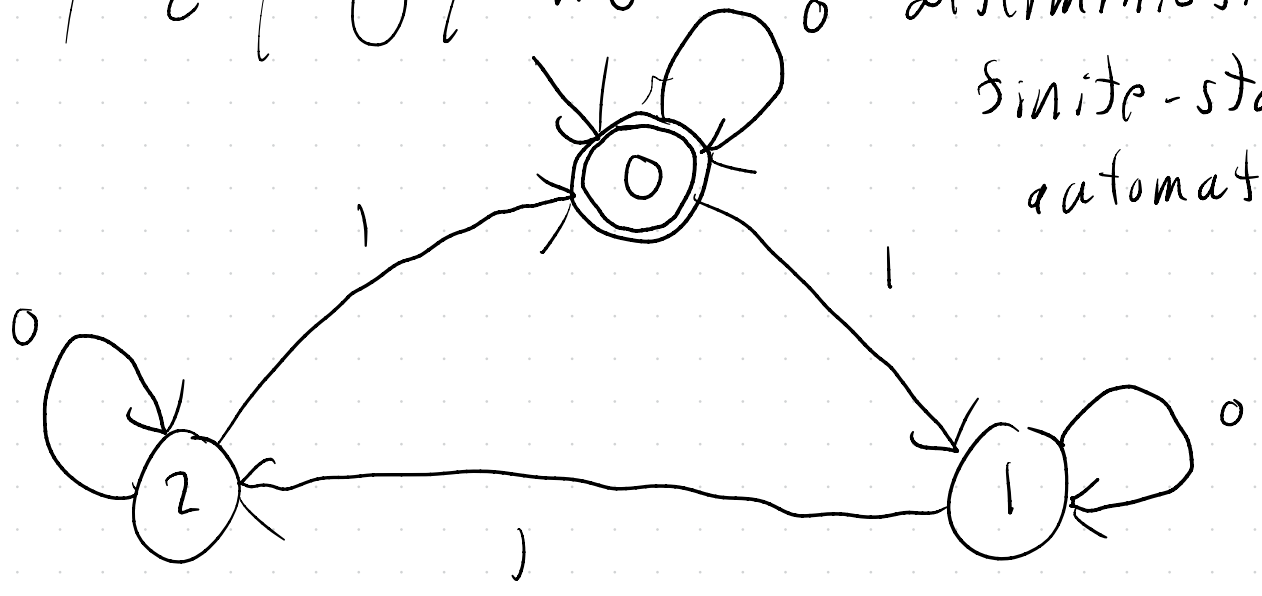
```
NumIsDivBy3(w[1..n])
    rem ← 0
    for i ← 1 to n
        if w[i] = 1
            rem ← (rem + 1) mod 3
    return (rem = 0)
```

	0	1	return true?
0	0	1	yes
1	1	2	no
2	2	0	no

states

finite-state machine

deterministic finite-state automata (DFA)



A DFA consists of five components:

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

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

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

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

$Q = \{0, 1, 2, 3\}$

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

$s = 0$

$A = \{0, 3\}$

extended transition function $\delta^* : Q \times \Sigma^* \to Q$

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

DFA accepts w iff $\delta^*(s, w) \in A$

rejects w otherwise

$$\delta^*(0, 01010110101) = \delta^*(0, 1010110101)$$

$= \delta^*(1, 0101101)$

$= \delta^*(1, 101101)$

$\vdots$

$= \delta^*(2, 01)$

$= \delta^*(2, 1)$

$= \delta^*(0, \varepsilon)$

$= 0$

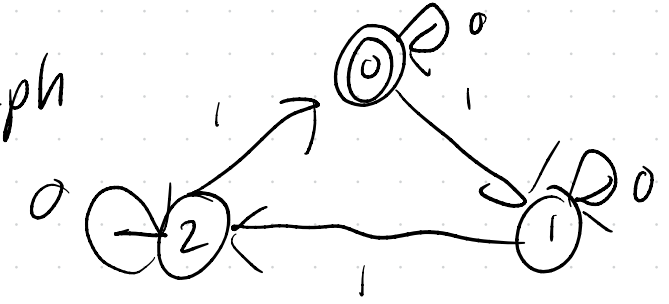
$\in A$

accept!

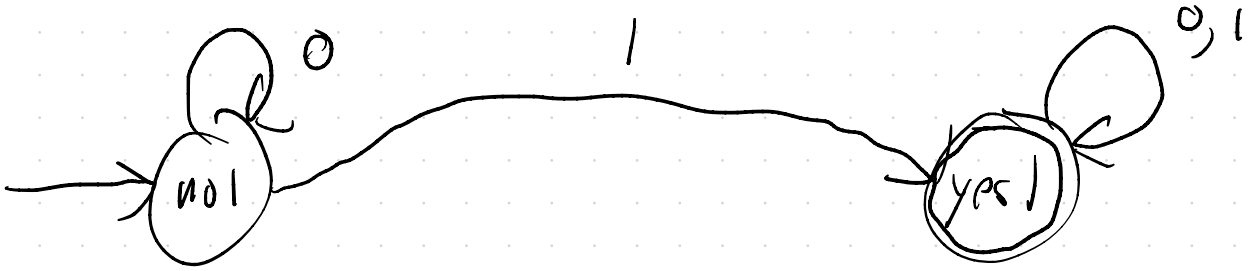
q	$\delta(q,0)$	$\delta(q,1)$	$q \in A$?
0	0	1	True
1	1	2	False
2	2	0	False

start at 0

or the graph



Binary strings contain a 1...



All strings with 11 as a substring:

```
CONTAINS11(w[1..n]):
  found ← FALSE
  for i ← 1 to n
      if i = 1
          last2 ← w[1]
      else
          last2 ← w[i−1]·w[i]
      if last2 = 11
          found ← TRUE
  return found
```

two variables: found a Boolean
last 2 a string of length ≤ 2

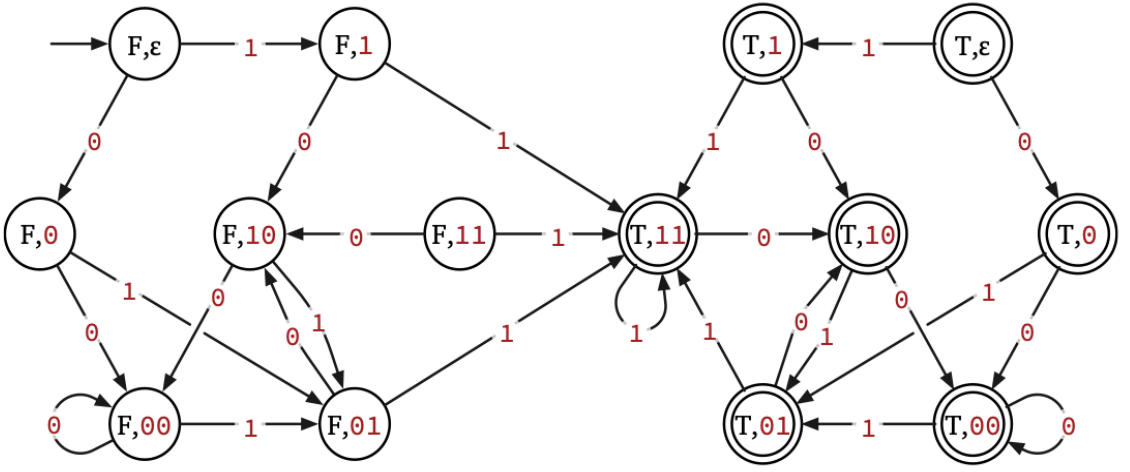
so $2 \cdot 7 = 14$

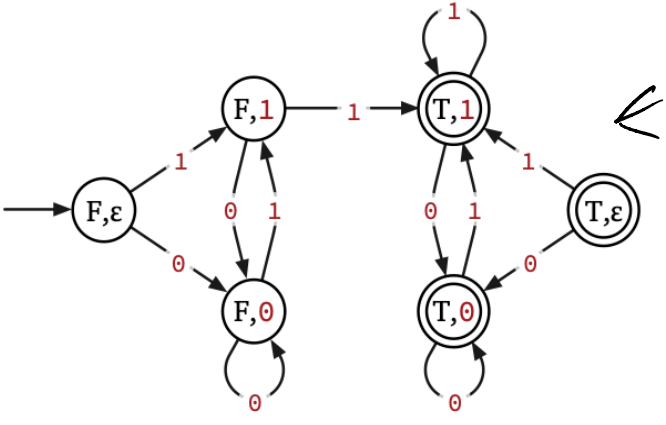
q	$\delta[q, 0]$	$\delta[q, 1]$
(False, ε)	(False, 0)	(False, 1)
(False, 0)	(False, 00)	(False, 01)
(False, 1)	(False, 10)	(**True**, 11)
(False, 00)	(False, 00)	(False, 01)
(False, 01)	(False, 10)	(**True**, 11)
(False, 10)	(False, 00)	(False, 01)
(False, 11)	(False, 10)	(**True**, 11)

q	$\delta[q, 0]$	$\delta[q, 1]$
(True, ε)	(True, 0)	(True, 1)
(True, 0)	(True, 00)	(True, 01)
(True, 1)	(True, 10)	(True, 11)
(True, 00)	(True, 00)	(True, 01)
(True, 01)	(True, 10)	(True, 11)
(True, 10)	(True, 00)	(True, 01)
(True, 11)	(True, 10)	(True, 11)

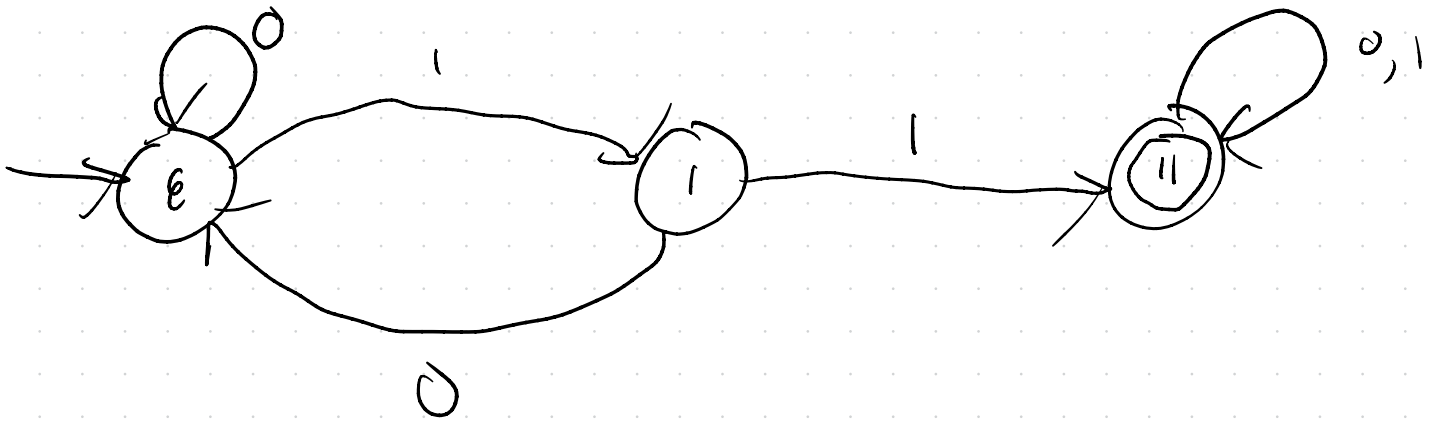
start at (False, ε)

accept (True, •)





only remember one last character



All numerals w that are a multiple of 5.

0110 1

x 2 +

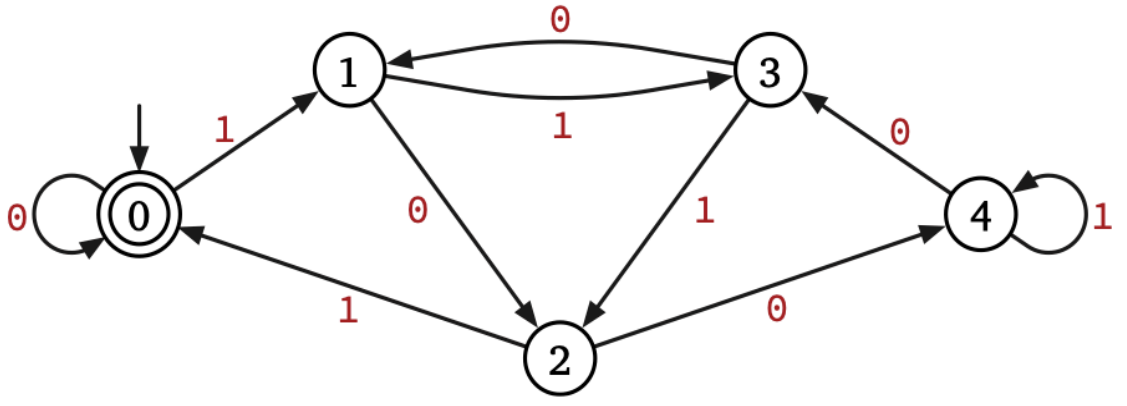
```
MULTIPLEOF5(w[1..n]):
  rem ← 0
  for i ← 1 to n
      rem ← (2 · rem + w[i]) mod 5
  if rem = 0
      return TRUE
  else
      return FALSE
```

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

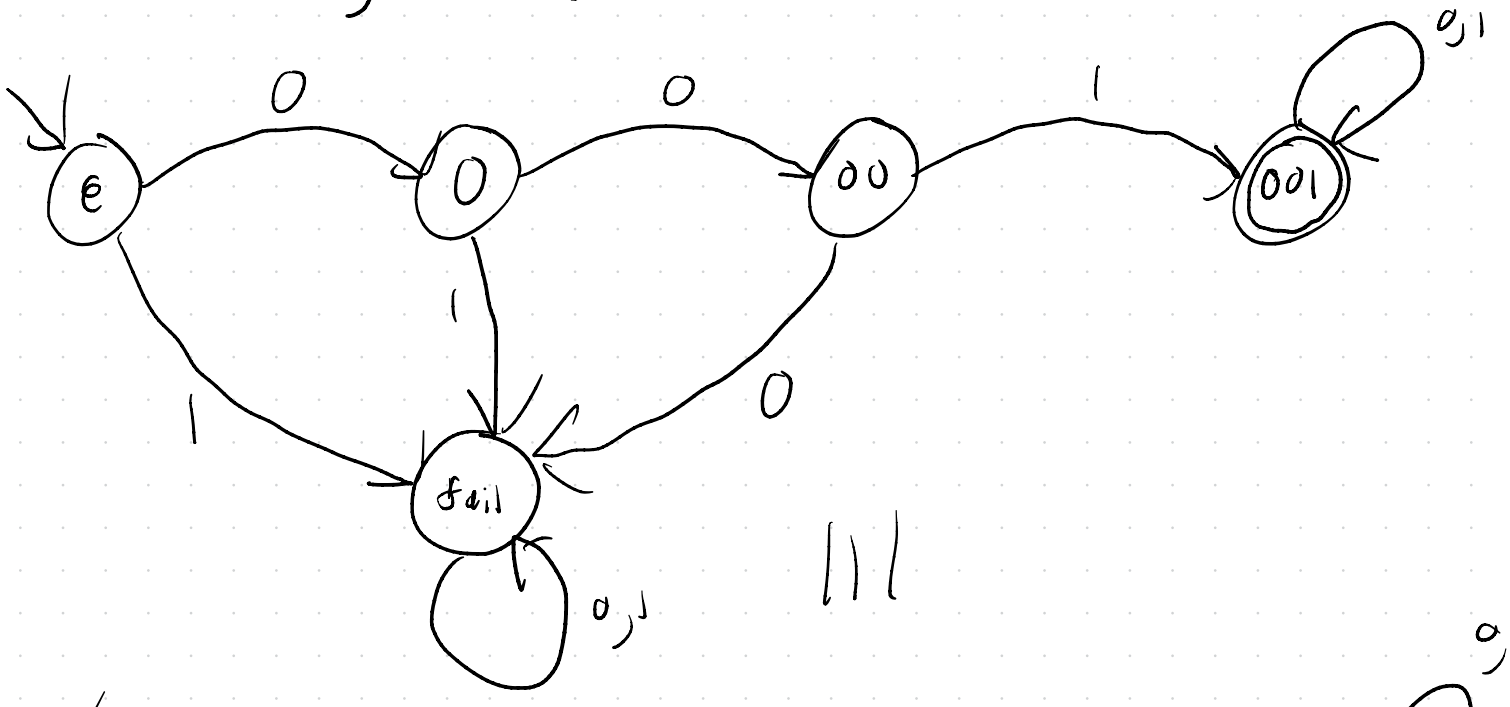
$Q = \{0, 1, 2, 3, 4\}$

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

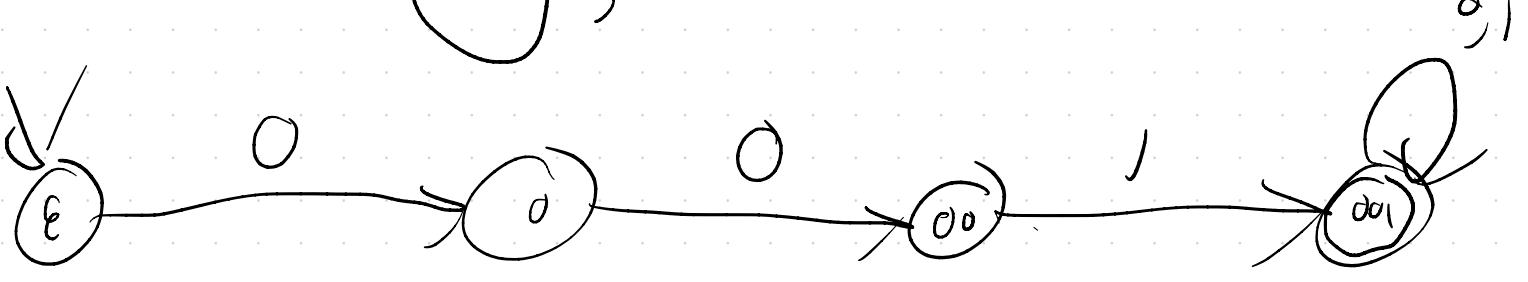
start at 0


0
1
2
3
4
0
1
0
1
0
1
1
0
0
1
1

All strings beginning with 001:



111



implicit fail if a transition does not exist. Write if you do this!