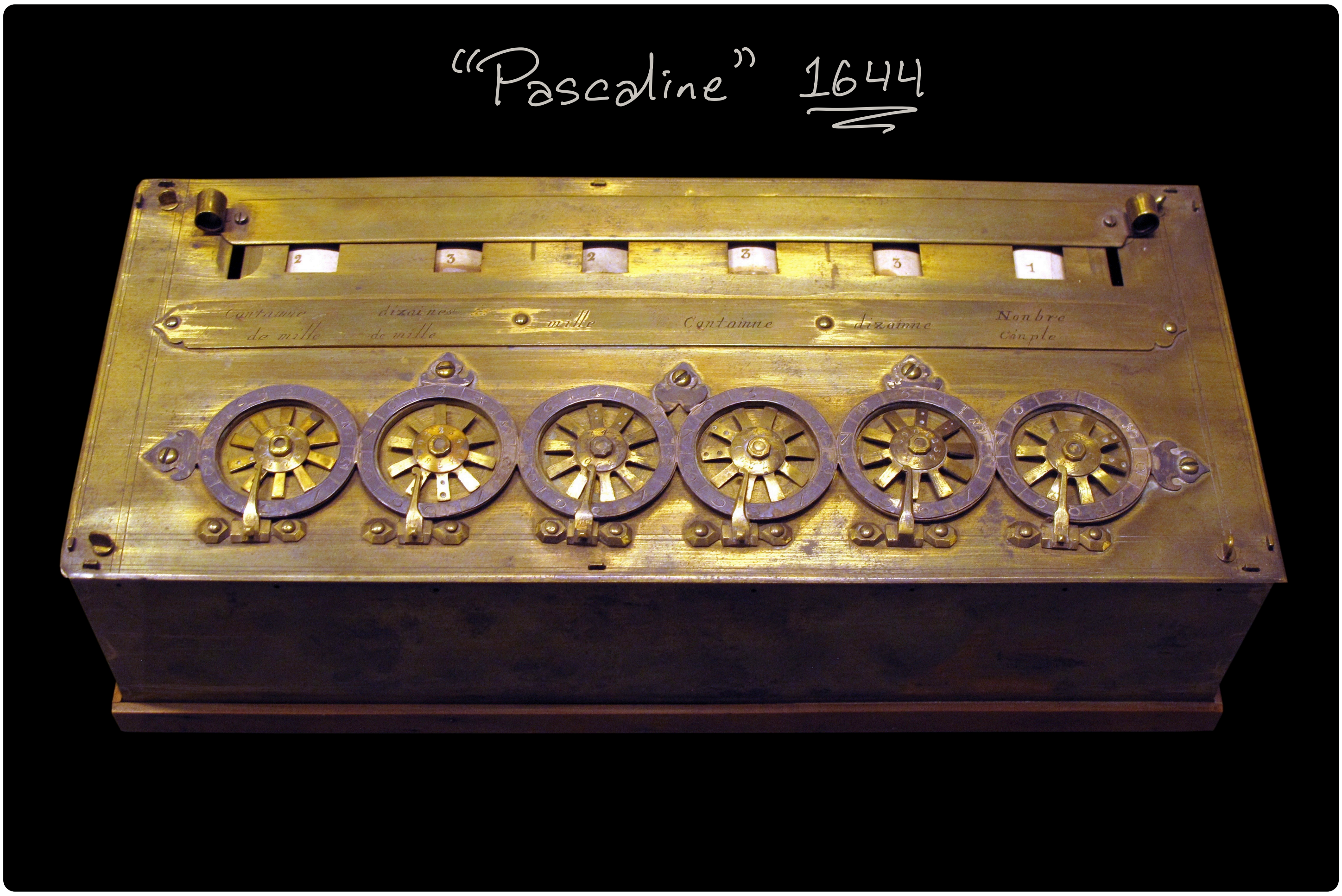
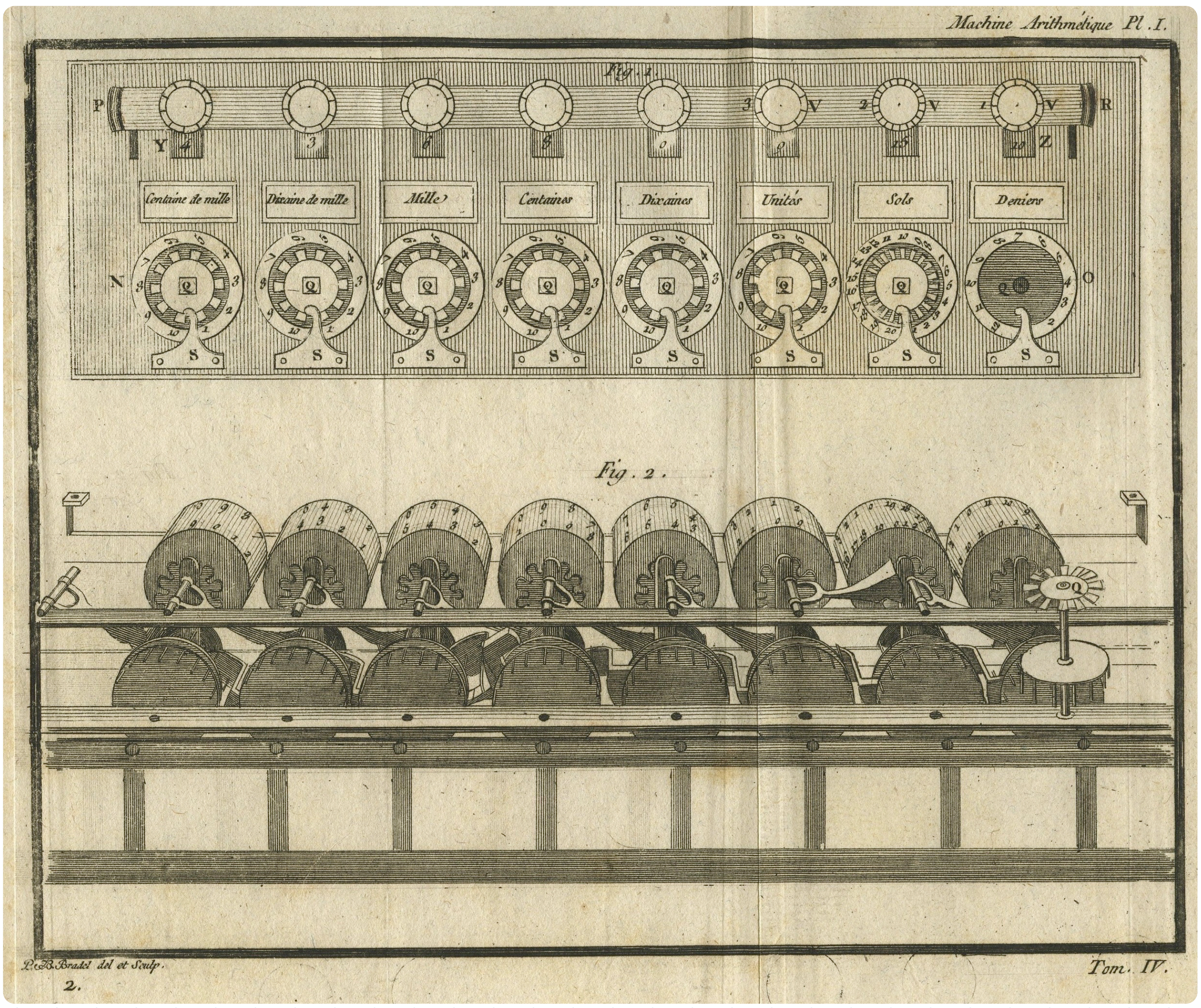
HW 0 due tonight

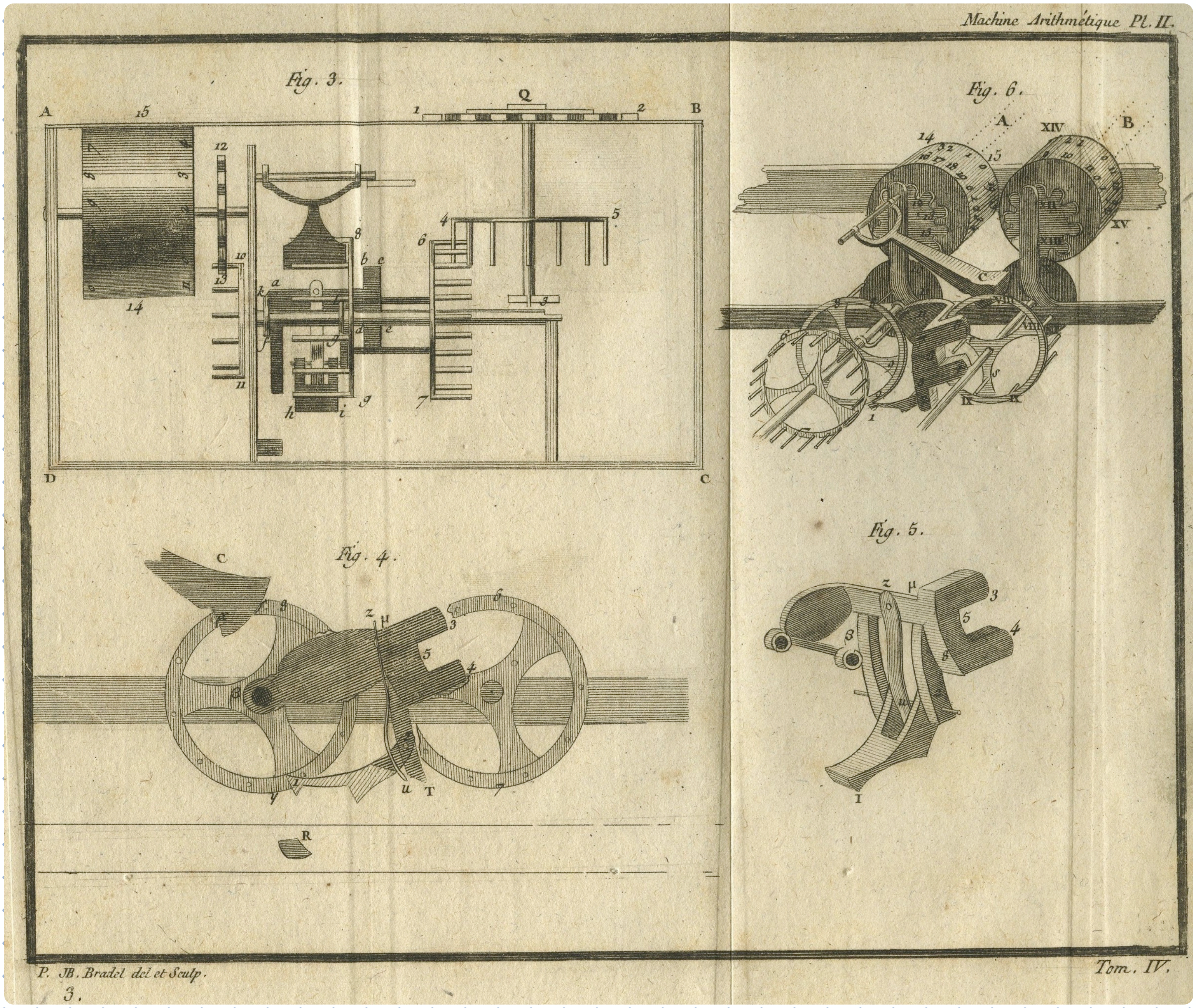
HW 1 due next Tuesday

Groups of up to 3 per problem (section A only)

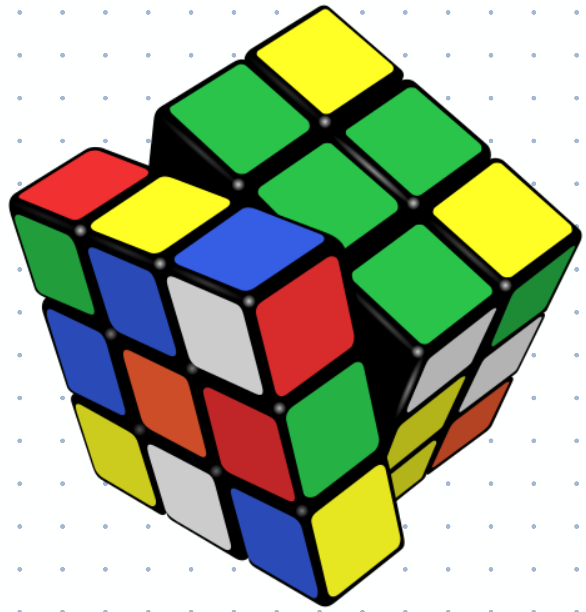
Jeff's OH WF 1-2:15 ish


"Pascaline" 1644


Machine Arithmétique Pl. I.
Fig. 1.
Centaine de mille
Dixaine de mille
Mille
Centaines
Dixaines
Unités
Sols
Deniers
Fig. 2.
Tom. IV.


Machine Arithmétique Pl. II.
Fig. 3.
Fig. 4.
Fig. 5.
Fig. 6.
P. JB. Bradel del et Sculp.
Tom. IV.
3.


CURTA

Deterministic finite-state automaton DFA

Finite-state machine FSA

Q — finite set of states

$s \in Q$ — start state

$A \subseteq Q$ — accepting states

Σ — input alphabet (finite set) $\{0,1\}$

$\delta : Q \times \Sigma \to Q$ — transition function

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

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

M accepts $w \iff \boxed{\delta^*(s,w) \in A}$

$L(M) = \{ w \in \Sigma^* \mid M \text{ accepts } w \}$

start

Is this binary # divisible by 5?

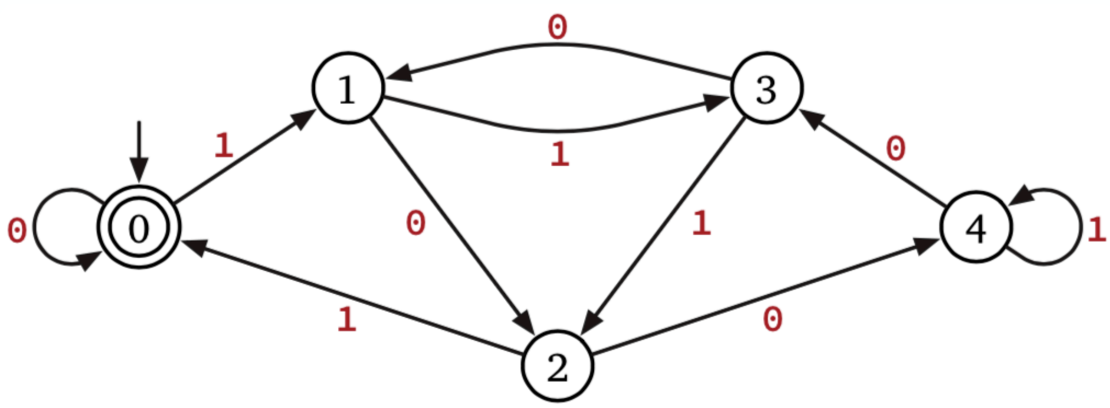
rem = STATE

```
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
```

transition

accepting

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



State-transition graph for MultipleOf5

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

DoSomethingCool(δ[ ][ ], A[ ], w[ ])

```
q ← 0
for i ← 1 to n
    q ← δ[q, w[i]]
return A[q]
```

$L = \{$ strings containing substring 11 $\}$

$= (0+1)^* 11 (0+1)^*$

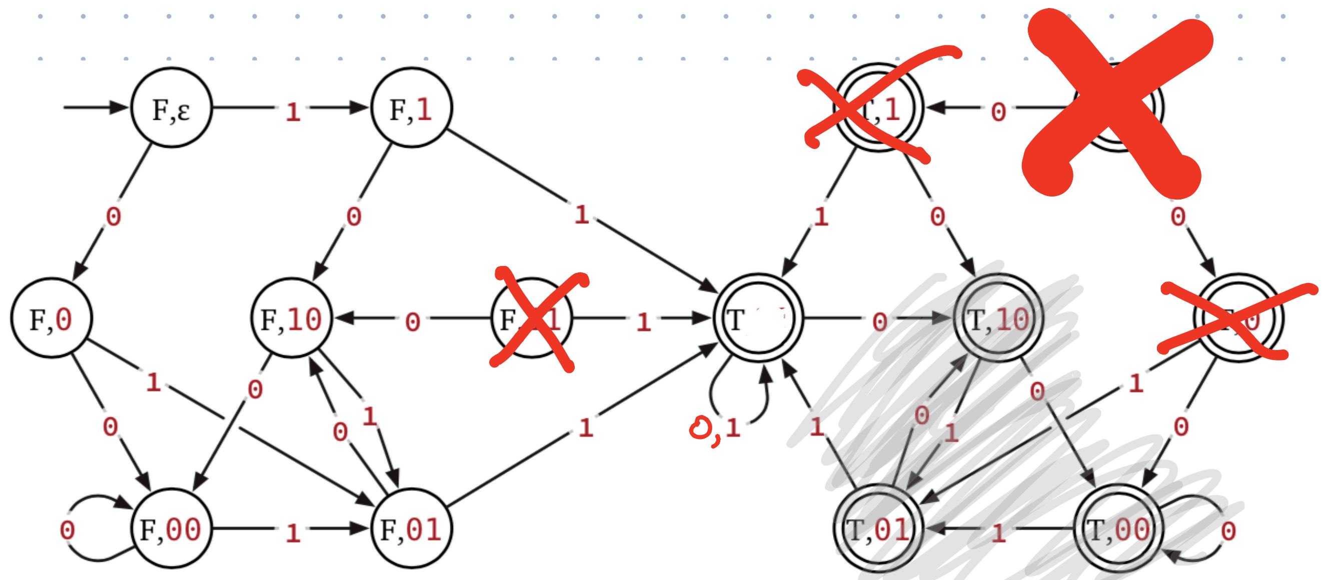
state = (found, last2)

2 × 7

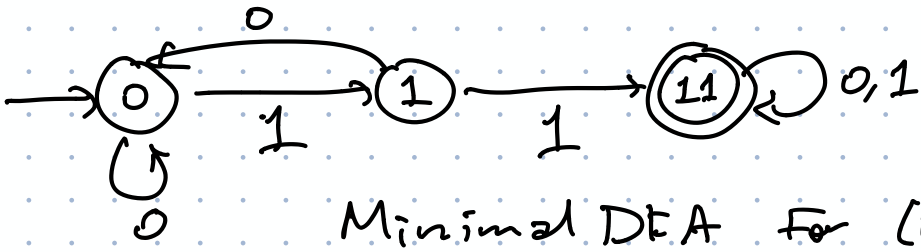
$\boxed{\text{14 states}}$

CONTAINS11($w[1..n]$):
  found ← FALSE ← *last2* ← ε
  for $i \leftarrow 1$ to n
    if $i = 1$
      last2 ← $w[1]$
    else
      last2 ← $w[i-1] \cdot w[i]$
    if *last2* = 11
      found ← TRUE
  return *found*

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



Our brute-force DFA for strings containing the substring 11



Minimal DFA for $(0+1)^*11(0+1)^*$

0: Last symbol read (if any) is 0, haven't seen 11

1: Last symbol read is 1, haven't seen 11

11: Have seen 11

$$\text{value}(w[1..n]) = \sum_i w[i]\cdot 2^{n-i}$$

$$\neq \sum_i w[i]\cdot 2^{i-1}$$

$$\text{binary}(w) = \begin{cases} 0 & w=\varepsilon \\ 2\cdot\text{binary}(x)+a & w=xa \end{cases}$$