Finite automata, automatic languages

1/13/21 10:29 AM

Administria.

- Remember to submit worksheets to Canvas.
- HW0 due this Friday (1/15).
- Schedule a (10–20 mins) meeting w/ me!

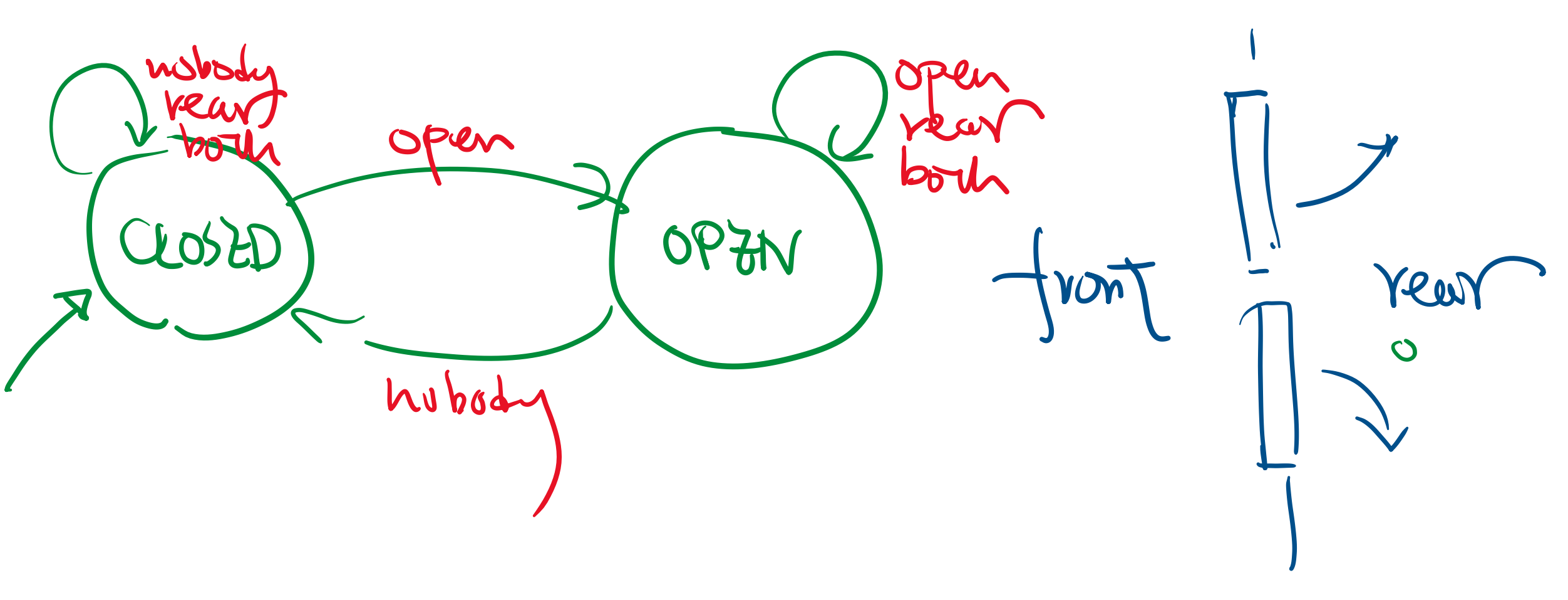
Question. What essential components to we need for a computer?

Ways to input/output. ~~Keyboard~~/switches, ~~monitor~~.

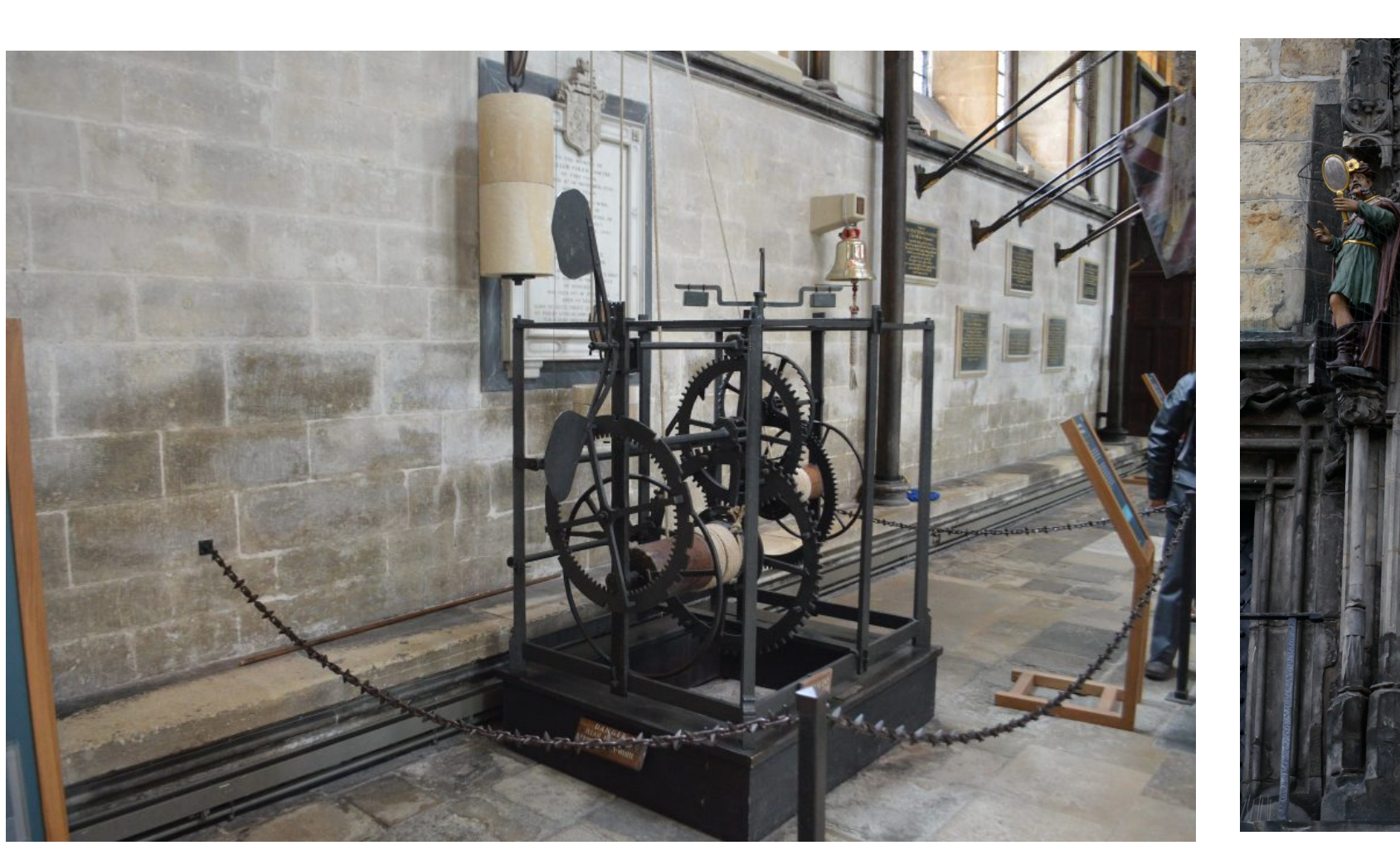
Storage. + access → ROM, RAM

processor + instructions, CPU.

Z1

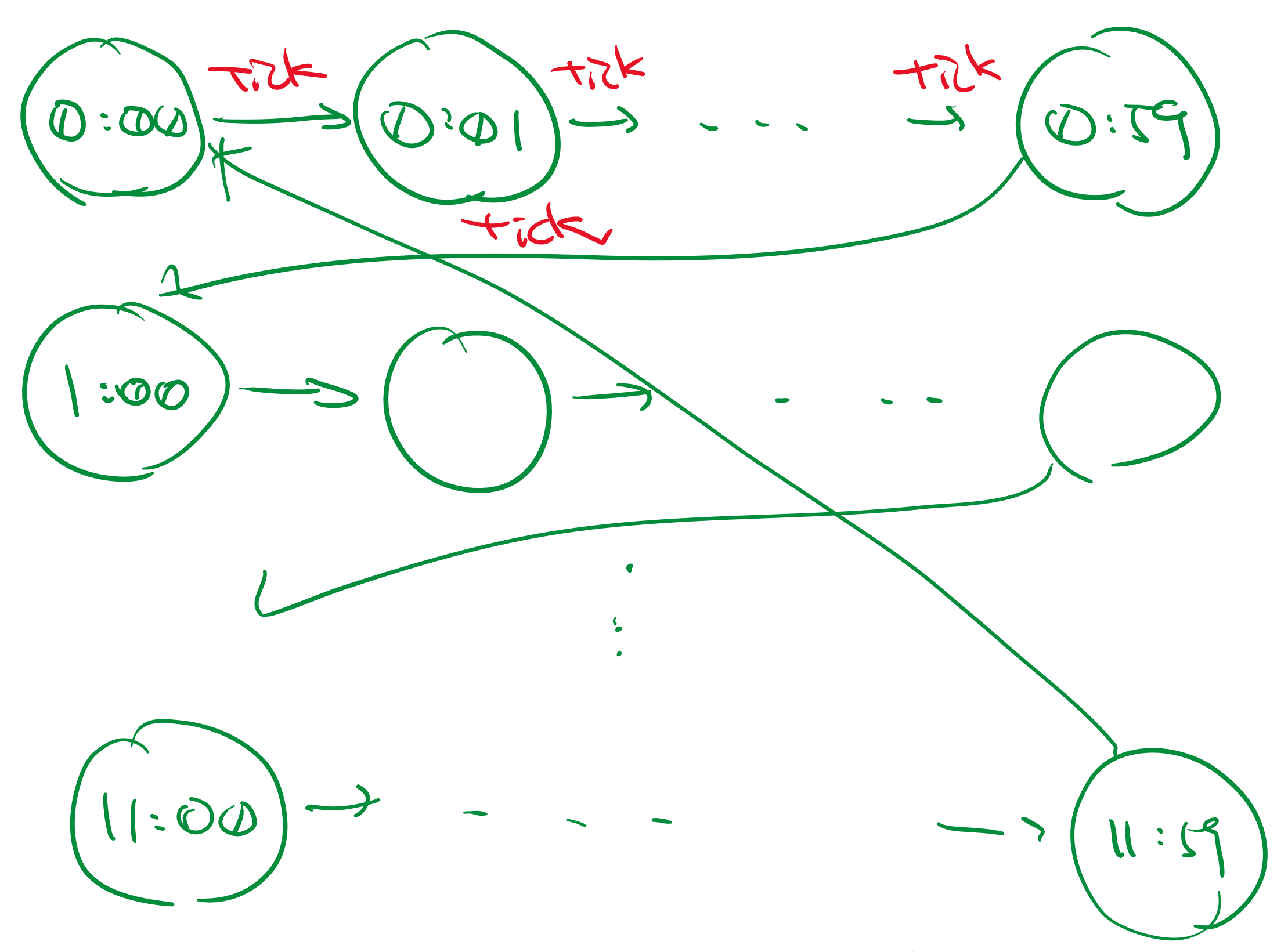


Z2



Salisbury Cathedral Mech. Clock [1386]

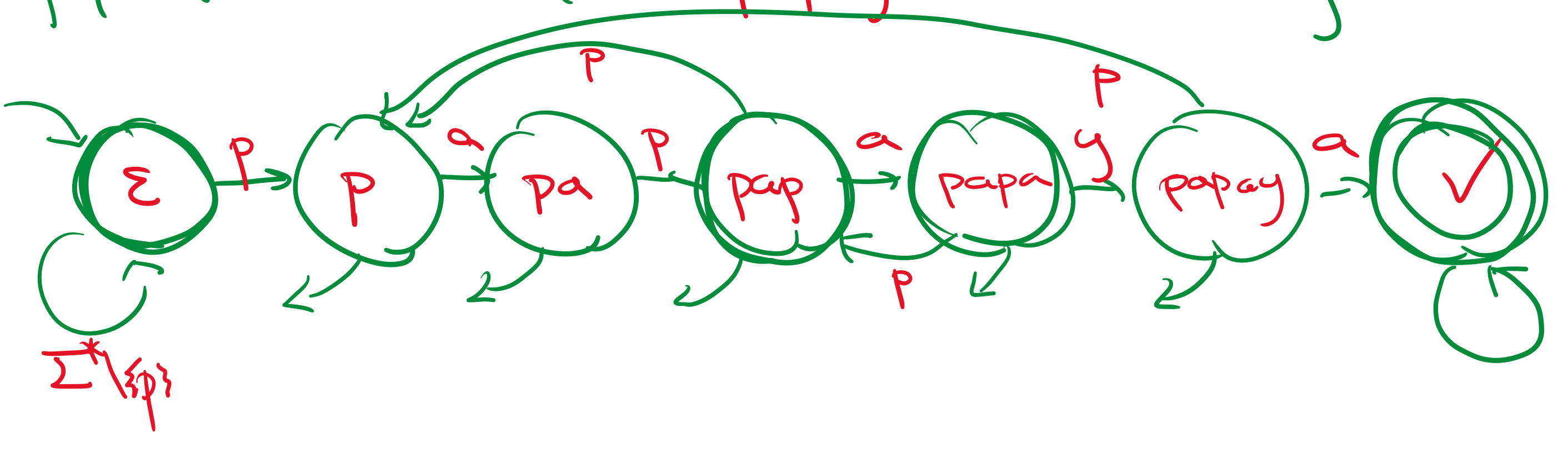
The Orloj [1410]



Z3. String matching automaton (KMP) [Knuth-Morris-Pratt 1970]

input: $w \in \Sigma^*$ $\Sigma = \{a \dots z\}$

output: Does w contains papaya as substring?



- Deterministic Finite-state Automaton (DFA)

States: Q finite in size

Start state: $s \in Q$

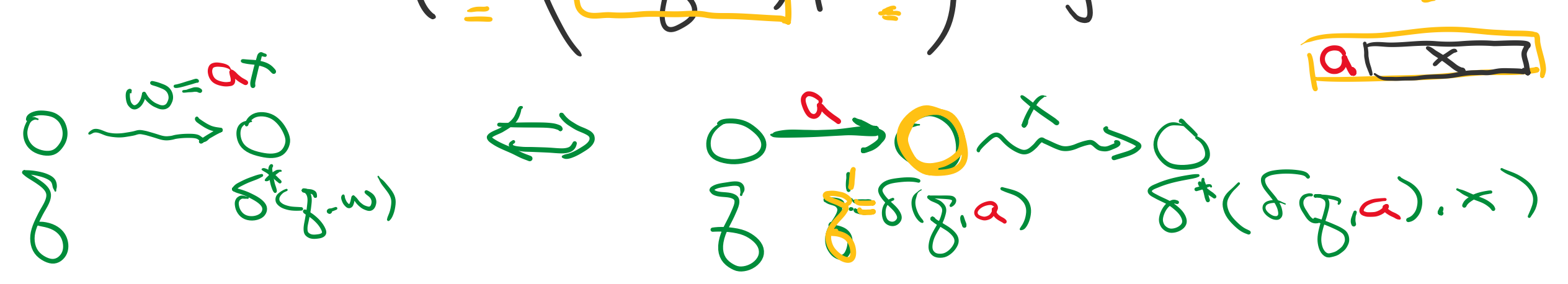
Accepting states: $A \subseteq Q$

Input alphabet: Σ

Transition function: $\delta : Q \times \Sigma \to Q$ $(q, a) \longmapsto q'$

$\delta^* : Q \times \Sigma^* \to Q$ extended Trans. fcn.

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



walking in the graph based on the input.

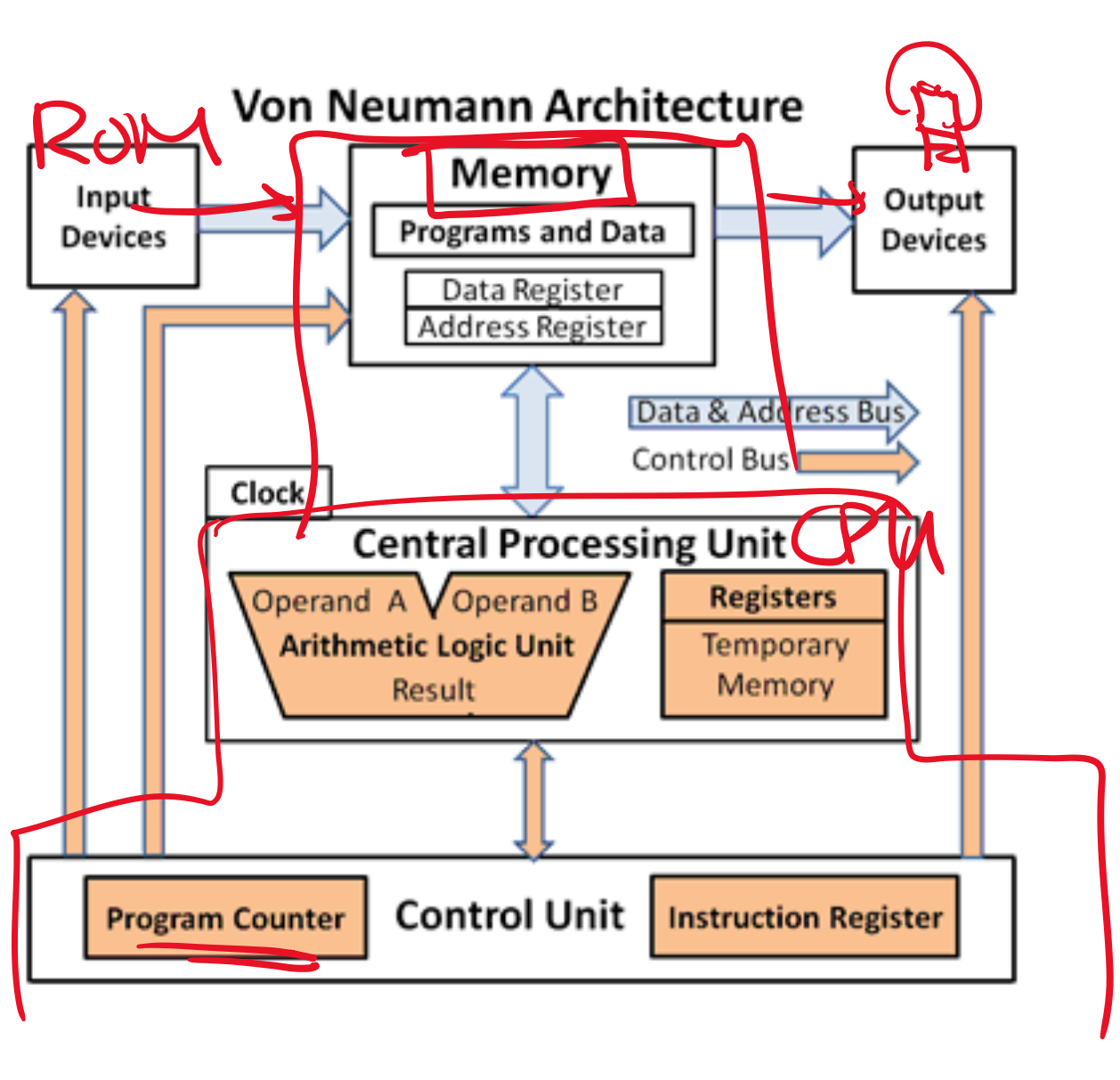
```
Accept?(M, w):
  input: DFA M=(Q, s, A, Σ, δ)
  output: does M accept w?

  $q \leftarrow s$
  for $i \leftarrow 1$ to $|w|$:
      $q \leftarrow \delta(q, w[i])$
  return $[q \in A]$
```

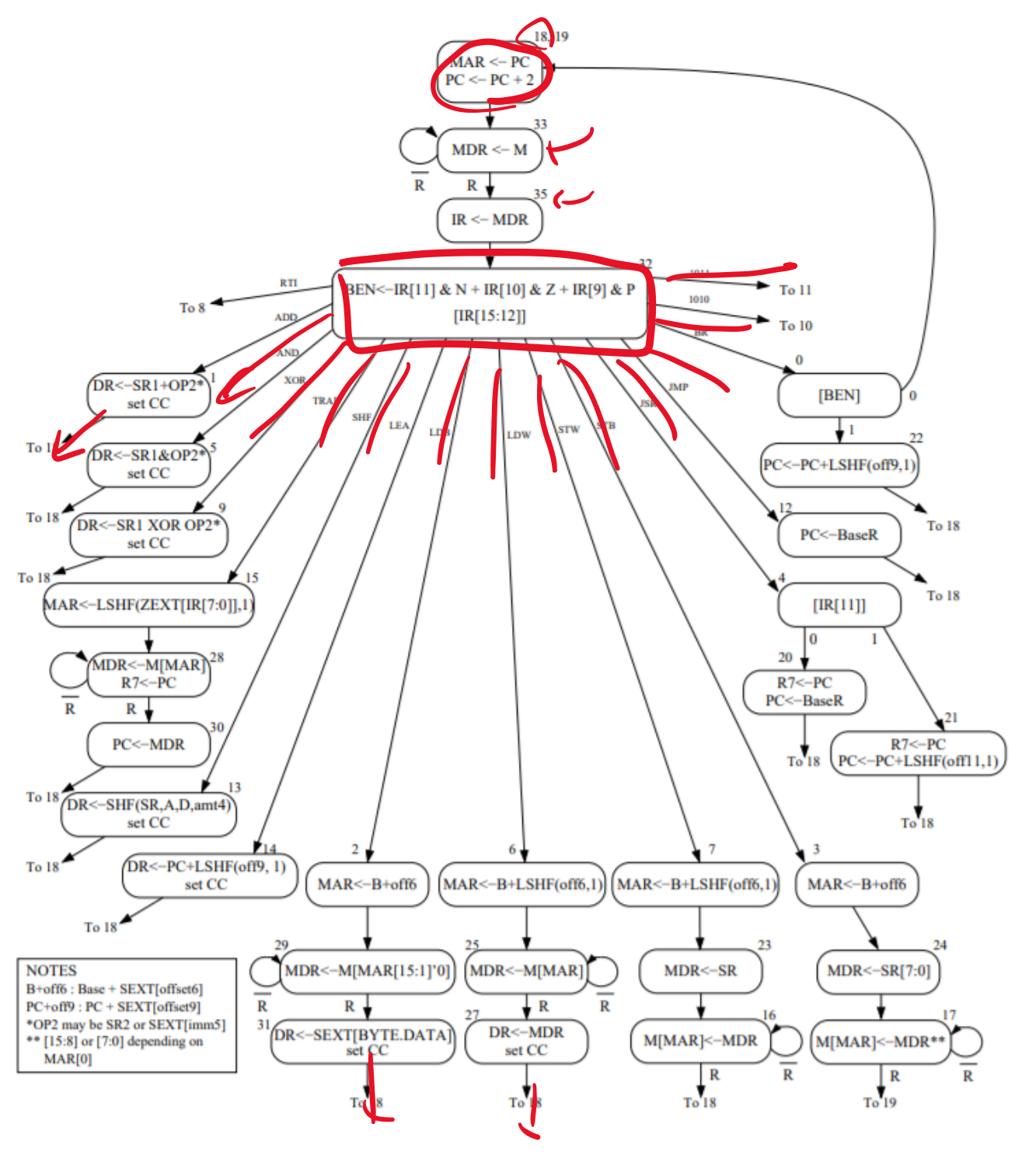
"DFA emulator"

not universal computation (yet).

Modeling/Philo.

Question 1. Does DFA really captures CPU...?





Theory

Question 2. What type of languages do DFAs compute?

Def. DFA M accepts w if $\delta^*(s, w) \in A$

language of M:

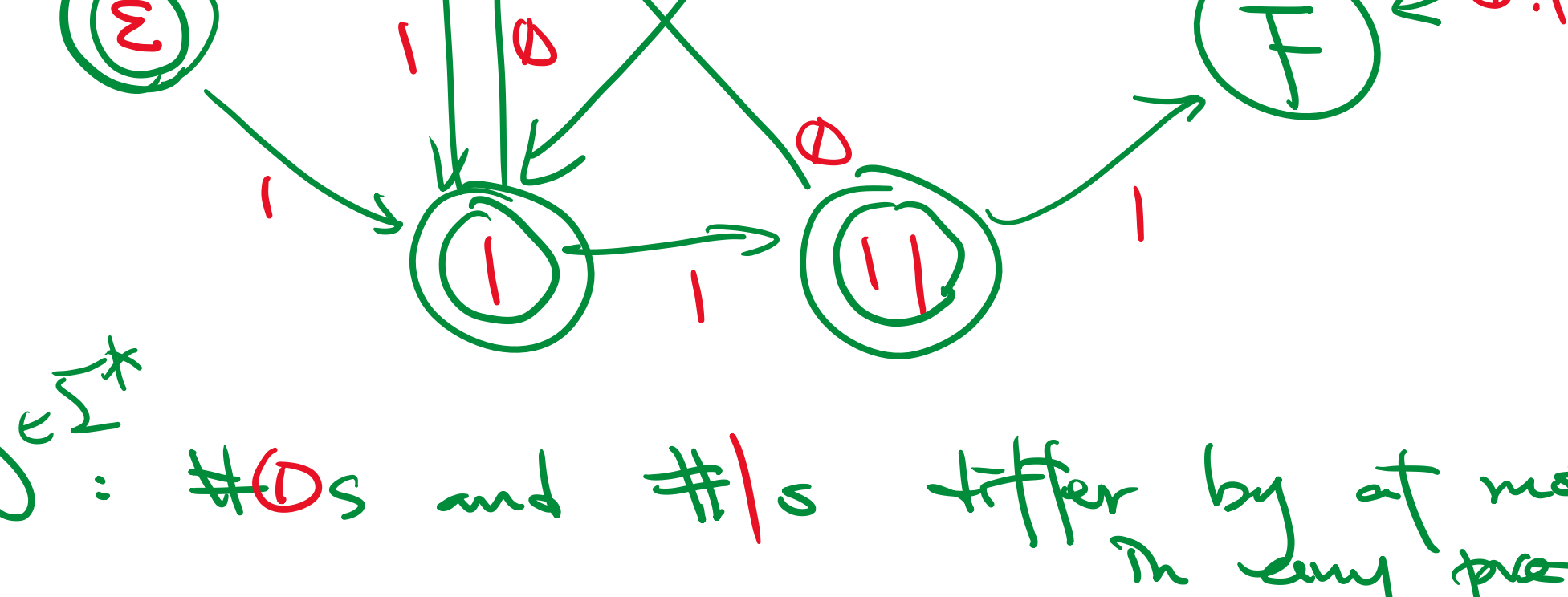
$$L(M) := \{ w \in \Sigma^* : \underbrace{M \text{ accepts } w}_{[\delta^*(s,w) \in A]} \}$$

Def. Languages accepted by some DFAs are call automatic.

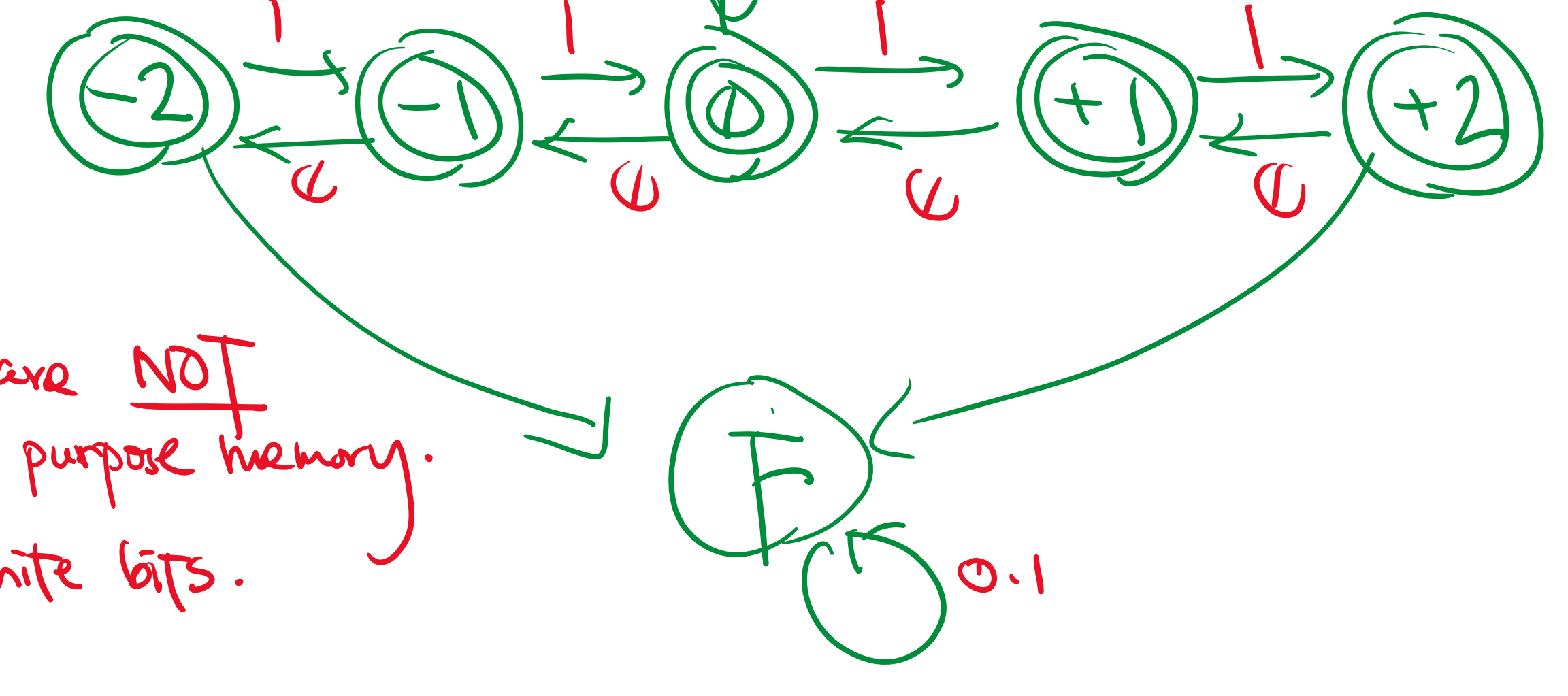
e.g. $L(KMP_{papaya}) = \{ w \in \Sigma_{eng}^* : w \text{ contains papaya as substring} \}$

$\Sigma^* \text{papaya} \Sigma^*$

$L_4 := \{ w \in \Sigma^* : w \text{ not containing } 000 \text{ nor } 111 \text{ as substring} \}$



$L := \{ w \in \Sigma^* : \#0\text{s and } \#1\text{s differ by at most 2 in any prefix of } w \}$



states are NOT general purpose memory. Only finite bits.

Question 3. DFA seems to be better than reg. exp. Are all regular languages automatic?

"Are machines w/ O(1) bits of memory better than model w/ no memory?"

Problem	Language
Programing language	Machine Models.
No-memory programs	Regular Expressions
O(1)-memory programs	DFA.
Algorithm	Instance of a model.