

Date : September 7, 2020

AFLL

In today's class :

- Equivalence of RE and FA
 - ①
 - ②
- RE to FA (Thompson Construction Method)
- FA to RE (State Elimination Method)
- Equivalence of two RE
(Do they represent the same language?)



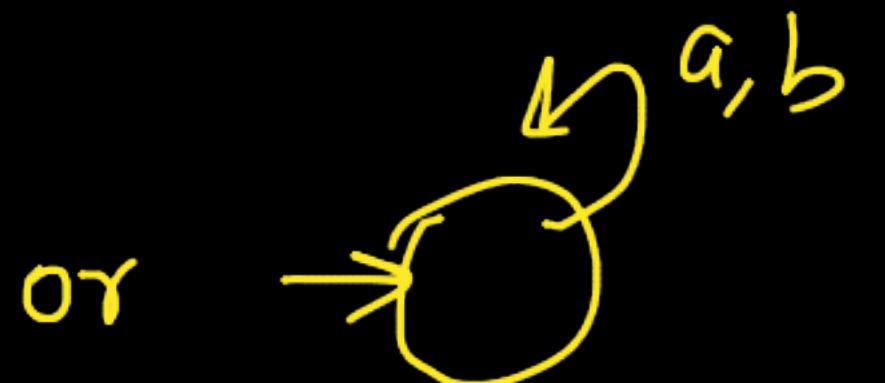
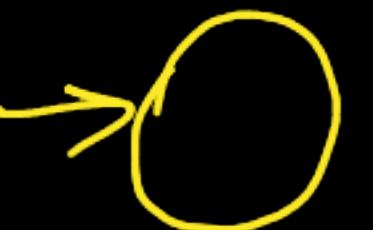
RE to FA (Thompson Construction method)

RE
↓
FA

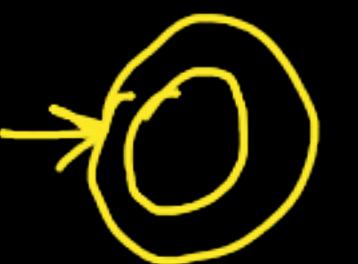
Base Cases

$\Sigma: \{a, b\}$

ϕ



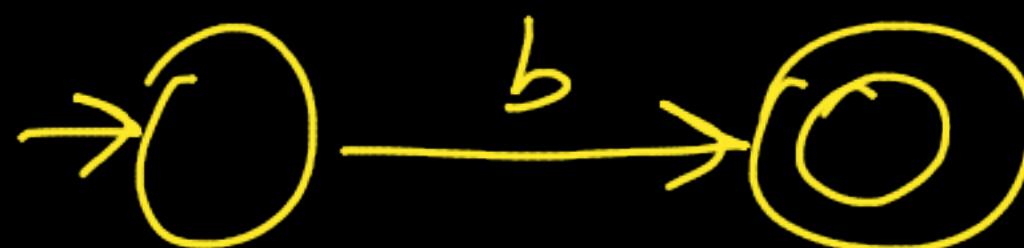
λ



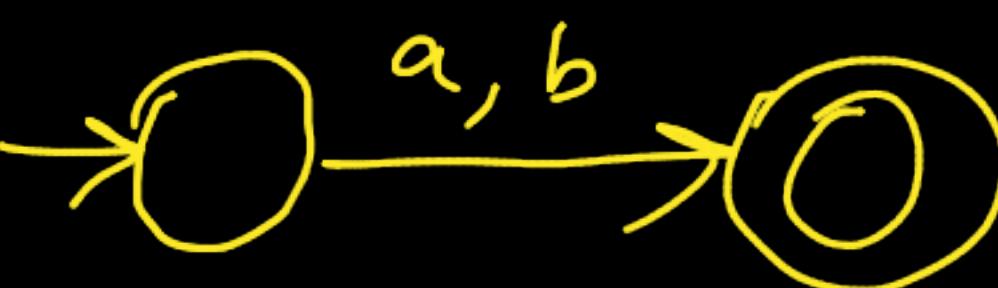
a



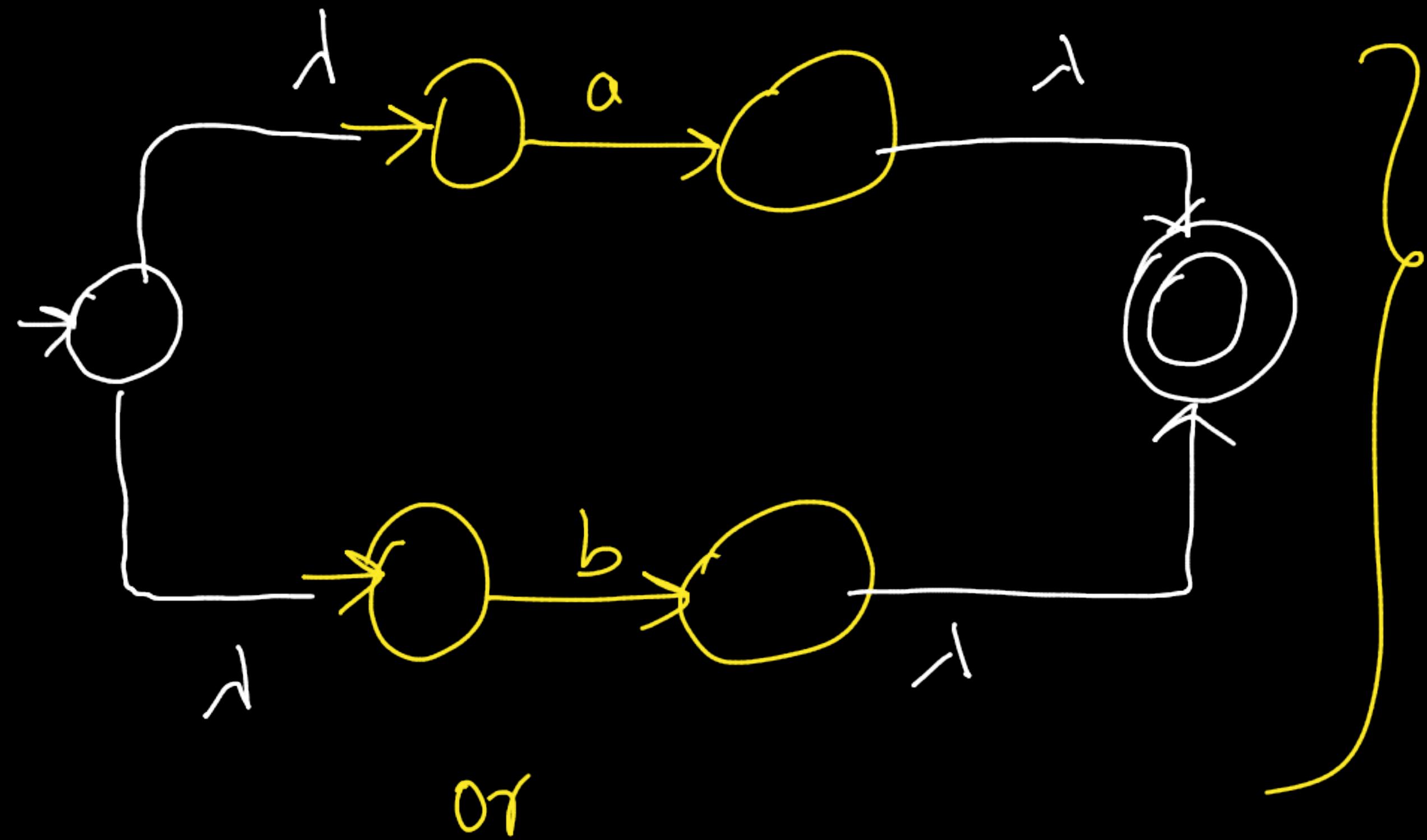
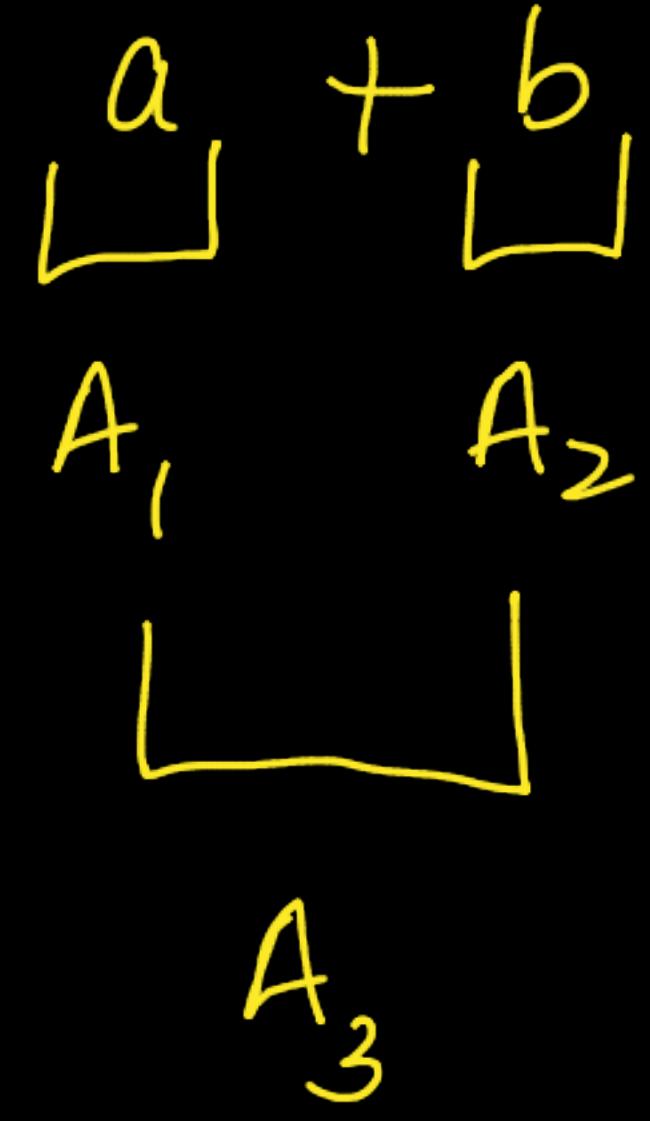
b



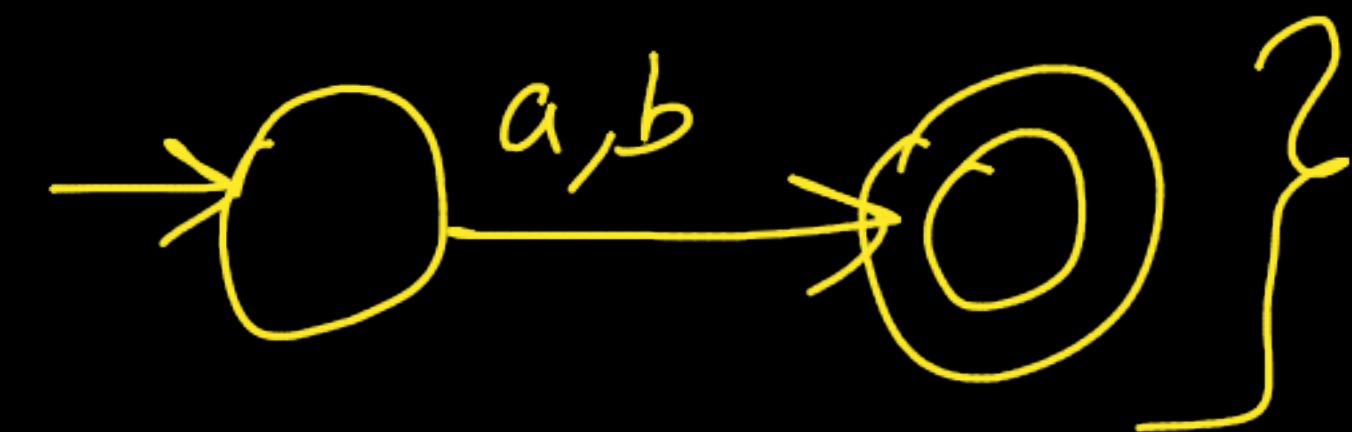
$a+b$ $\mathcal{L} = \{a, b\}$



Union

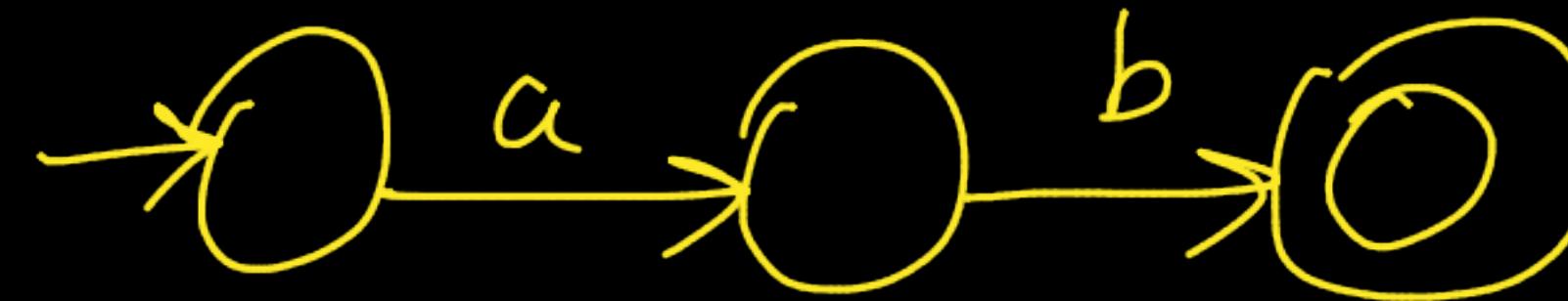


or

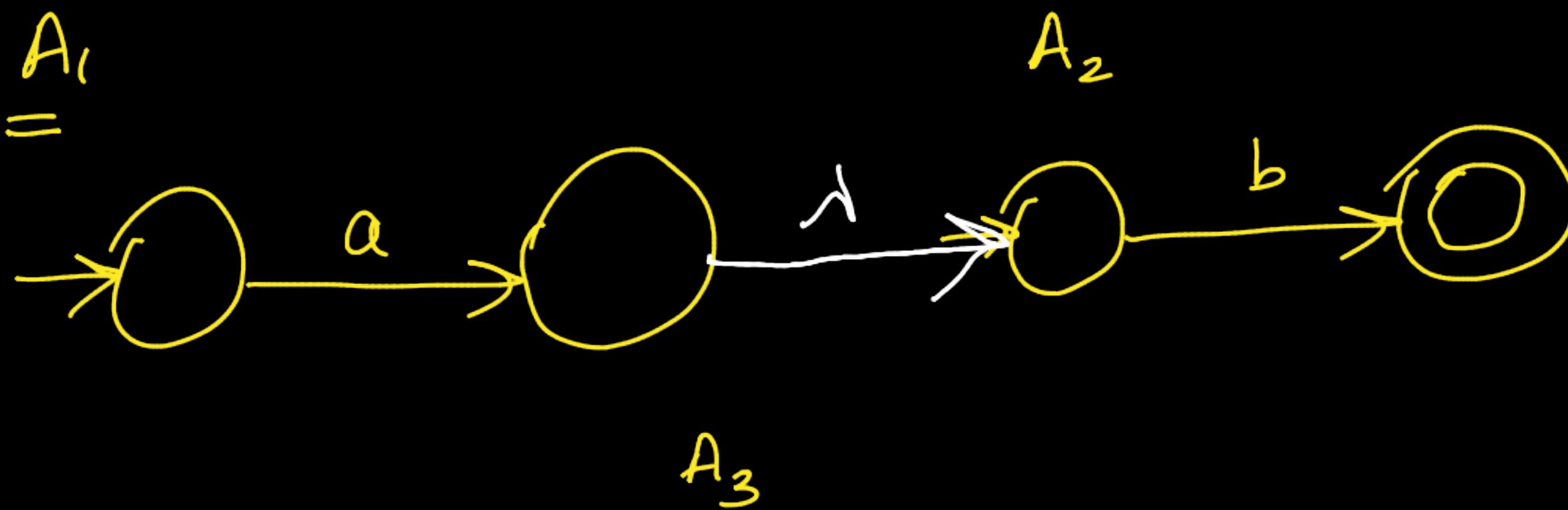


Concatenation

$$ab = \underline{A_1} \underline{\underline{A_2}} = \underline{\underline{\underline{A_3}}}$$



or



Kleene Star or a Kleene Closure

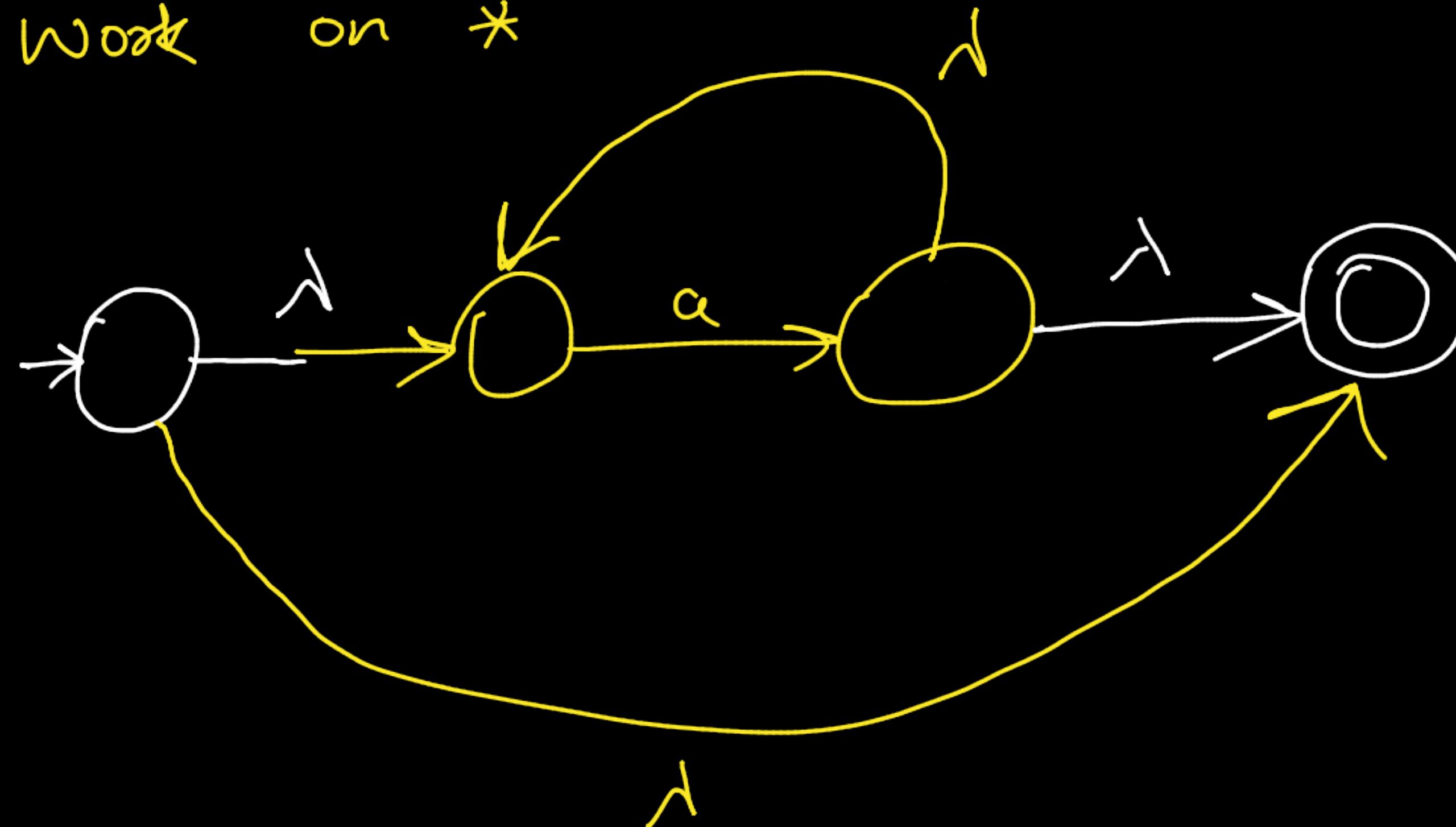
R^*

- 1) Create an automata to accept R
- 2) Work on *

Example

a^*

1)



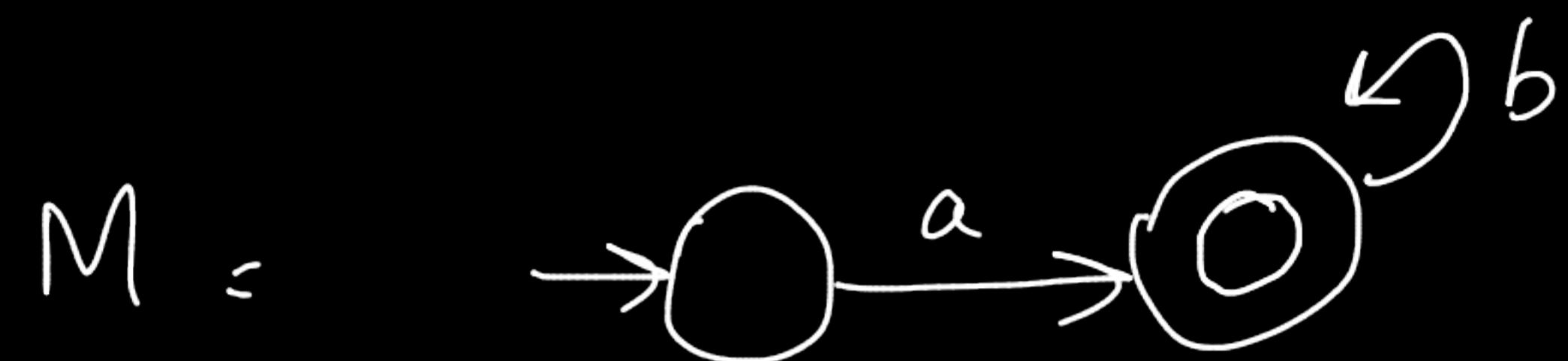
or



Examples to convert RE to FA (NFA/ λ -NFA)

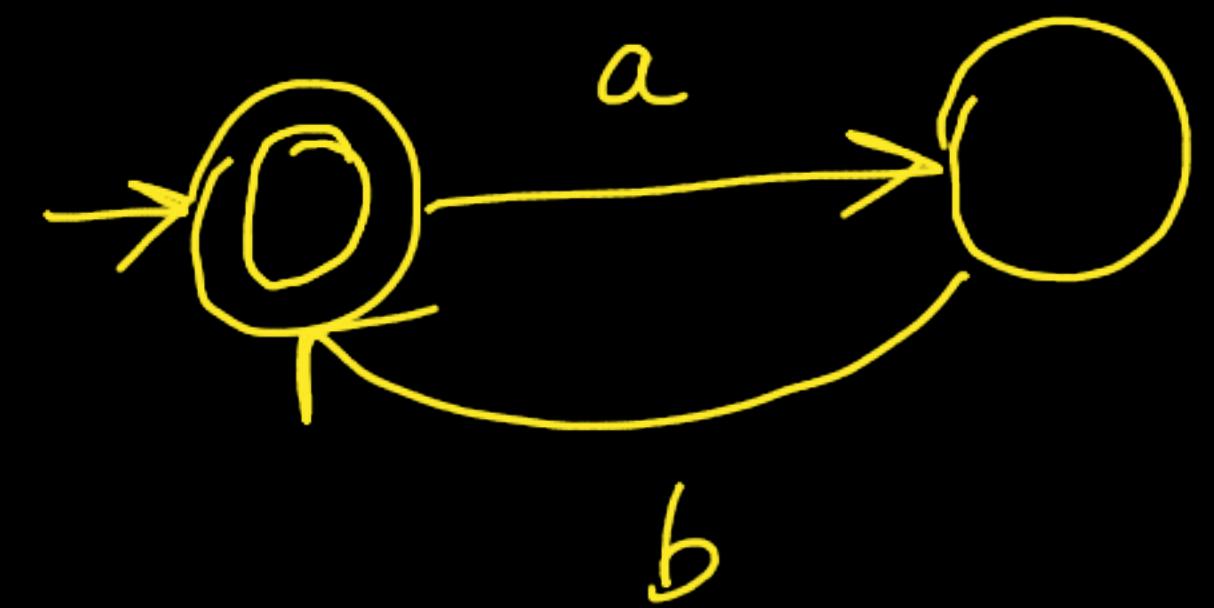
① $R = ab^*$

Lang = { String starts with an 'a'
followed by any no. of b 's
0 or more b 's }



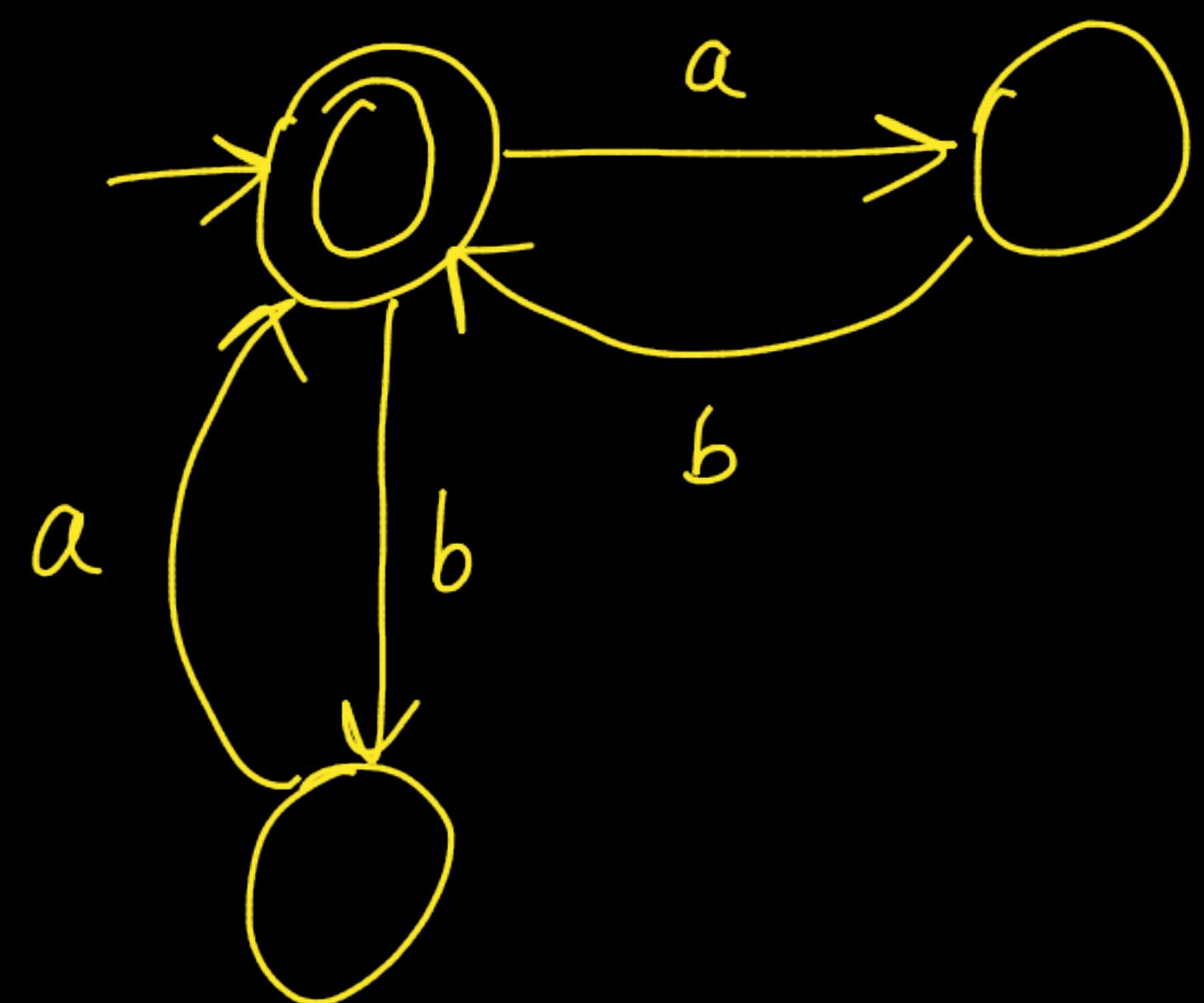
$$L(R) = L(M)$$

② $(\underset{=}{{ab}})^*$
 $L = \{\lambda, \underline{ab}, abab, ababab, abababab, \dots\}$

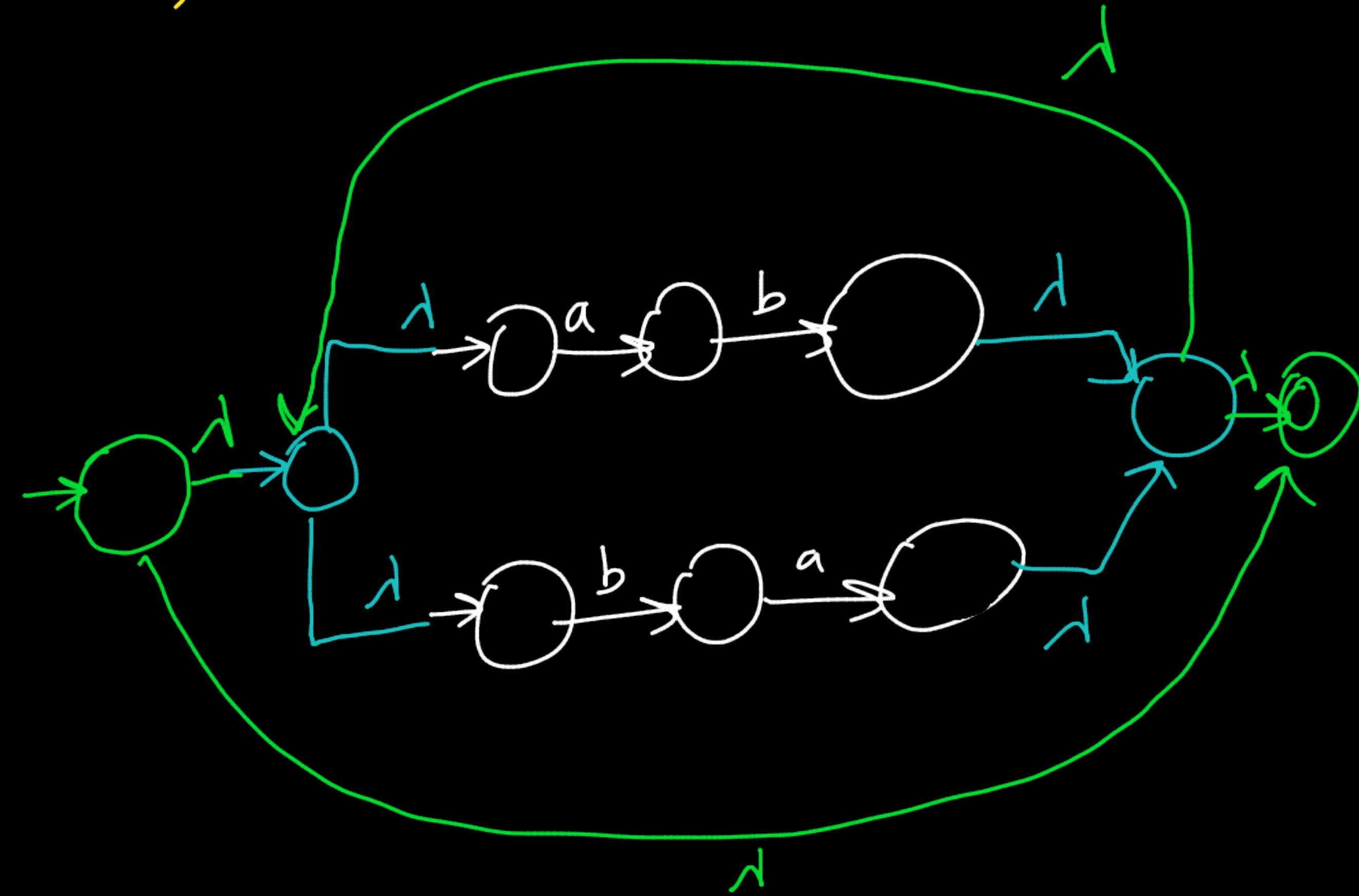


Intuitive soln.

③ $(\underline{ab} + \underline{ba})^*$
 $\mathcal{L} = \{\lambda, ab, ba, ababab, bababa, \underline{\overline{abbaab}}, baabab, \dots\}$



Intuitive way



Thompson Construction

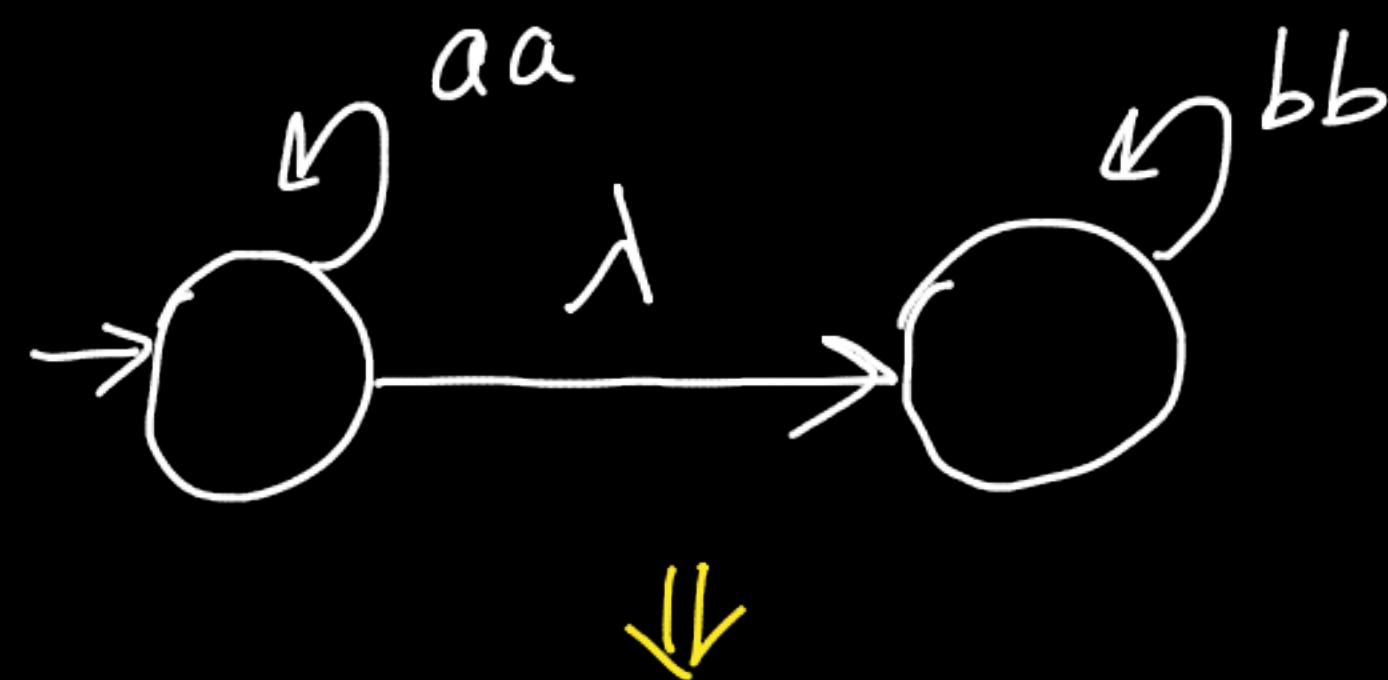
④

$$\underline{(aa)^* \cdot (bb)^*} + \underline{a(aa)^* b(bb)^*}$$

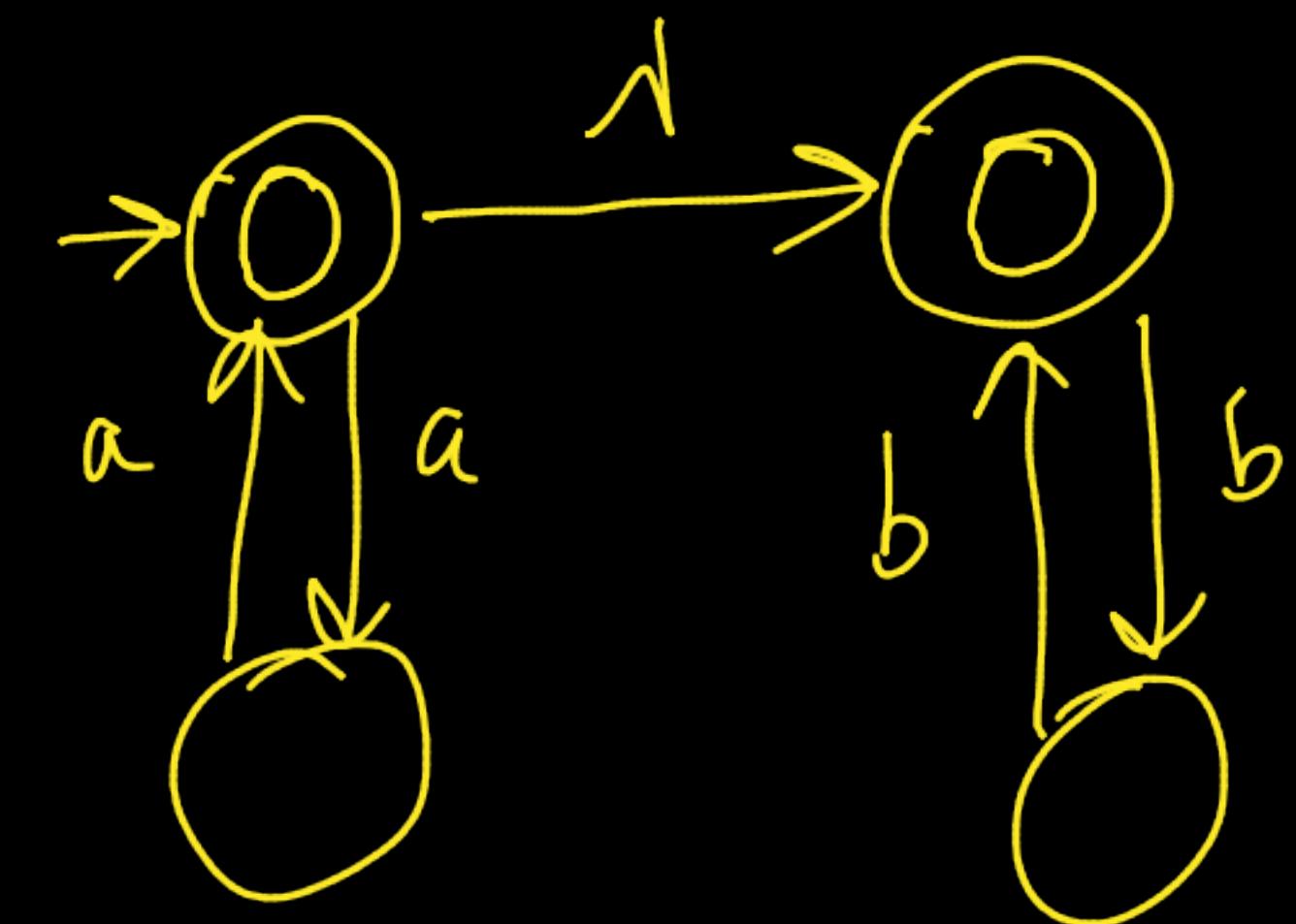
GTG

Generalized Transition Graphs

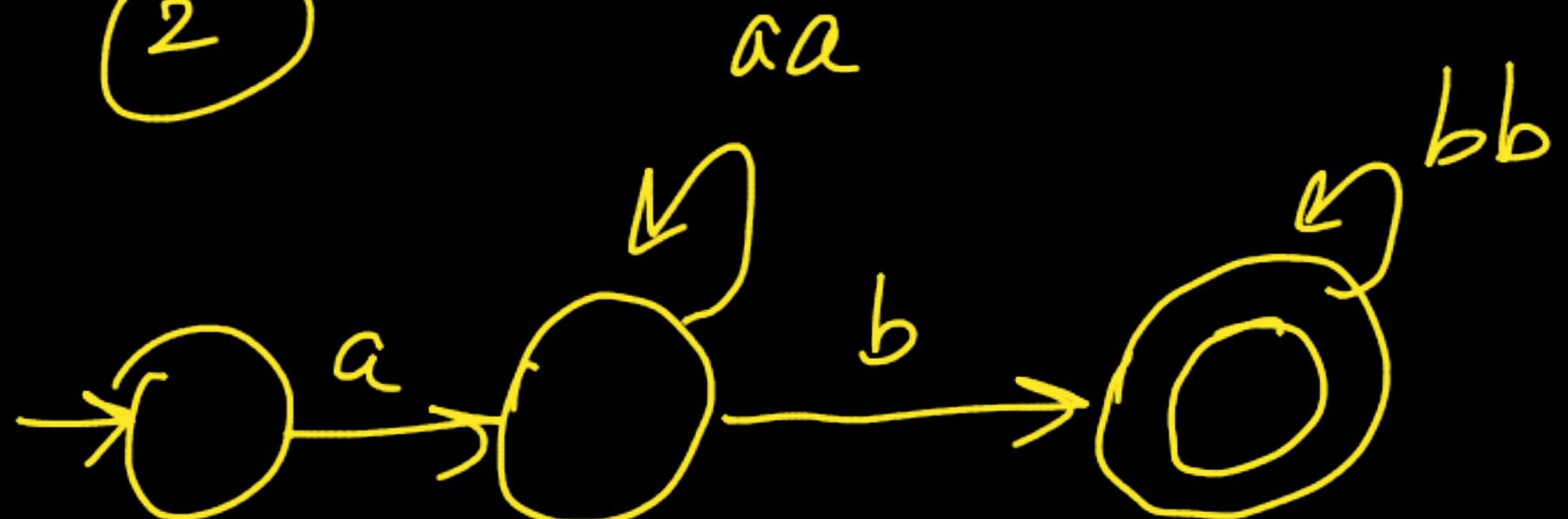
①



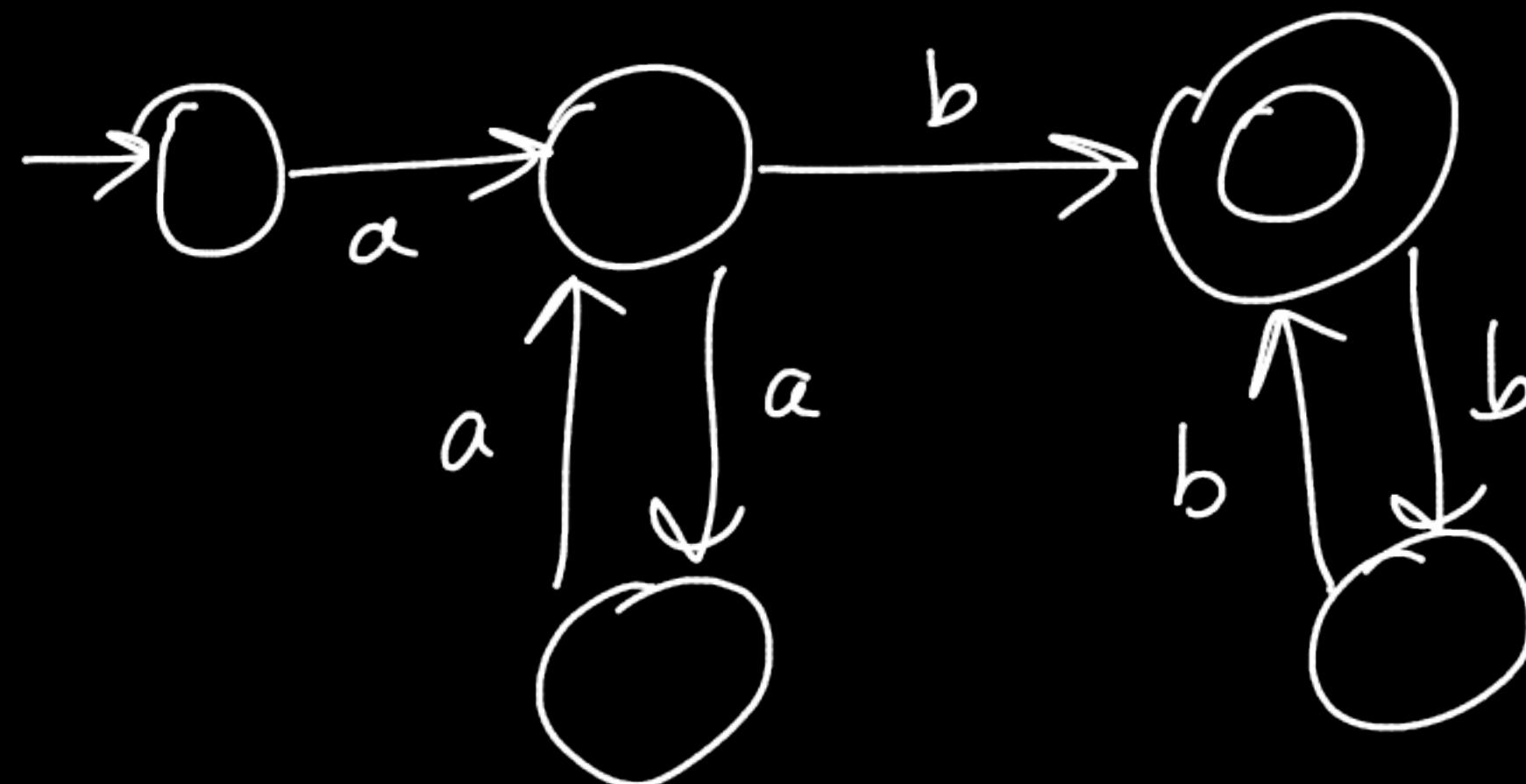
↓



②



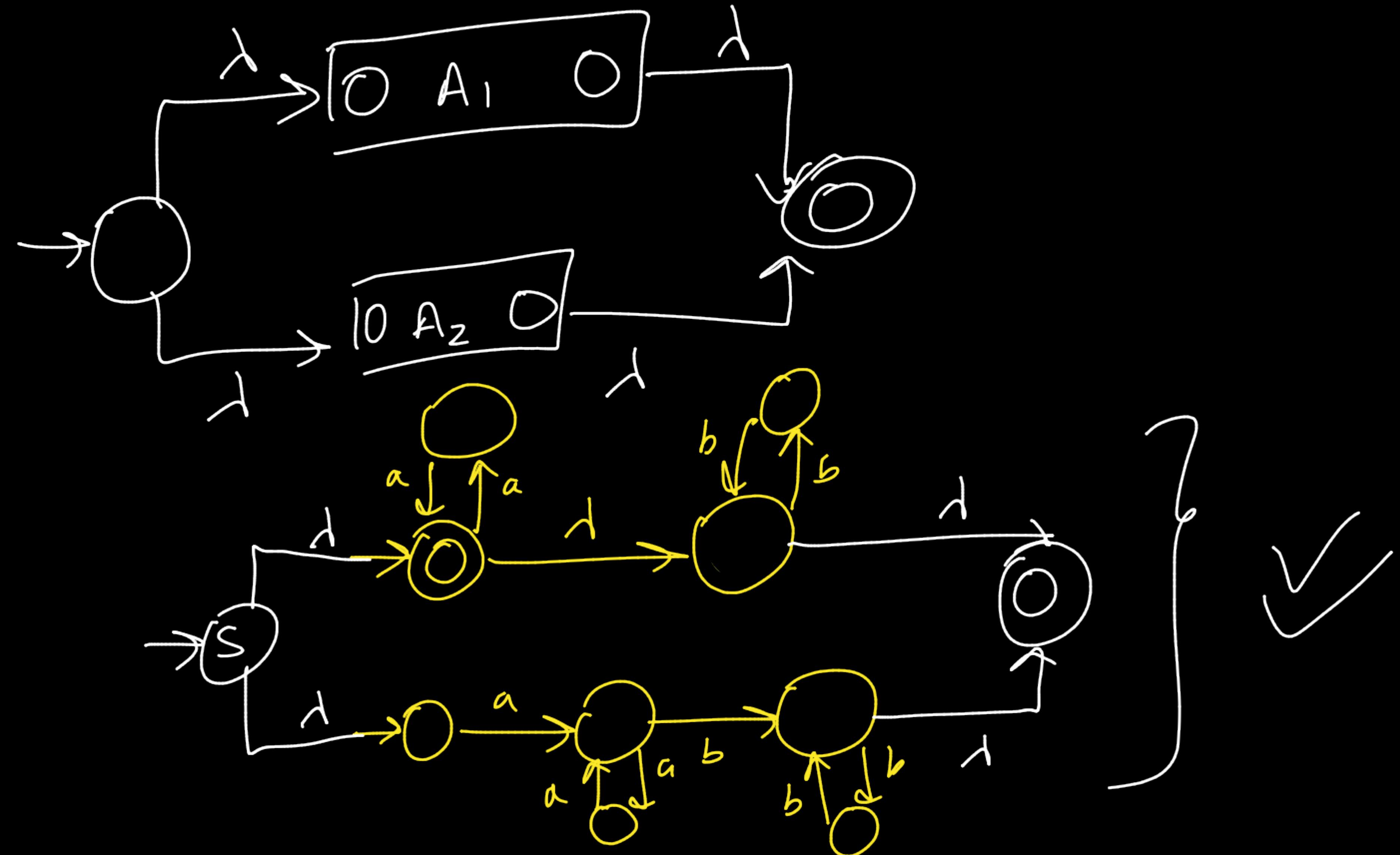
↓



4

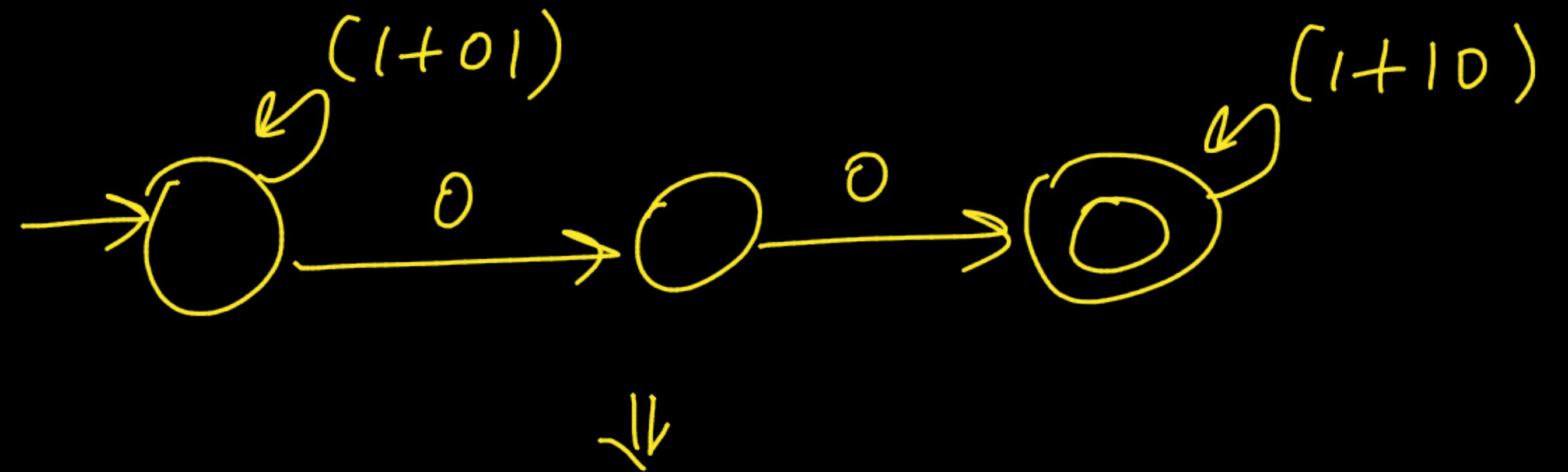
Continued

$$(aa)^* (bb)^* + a(aa)^* b(bb)^*$$

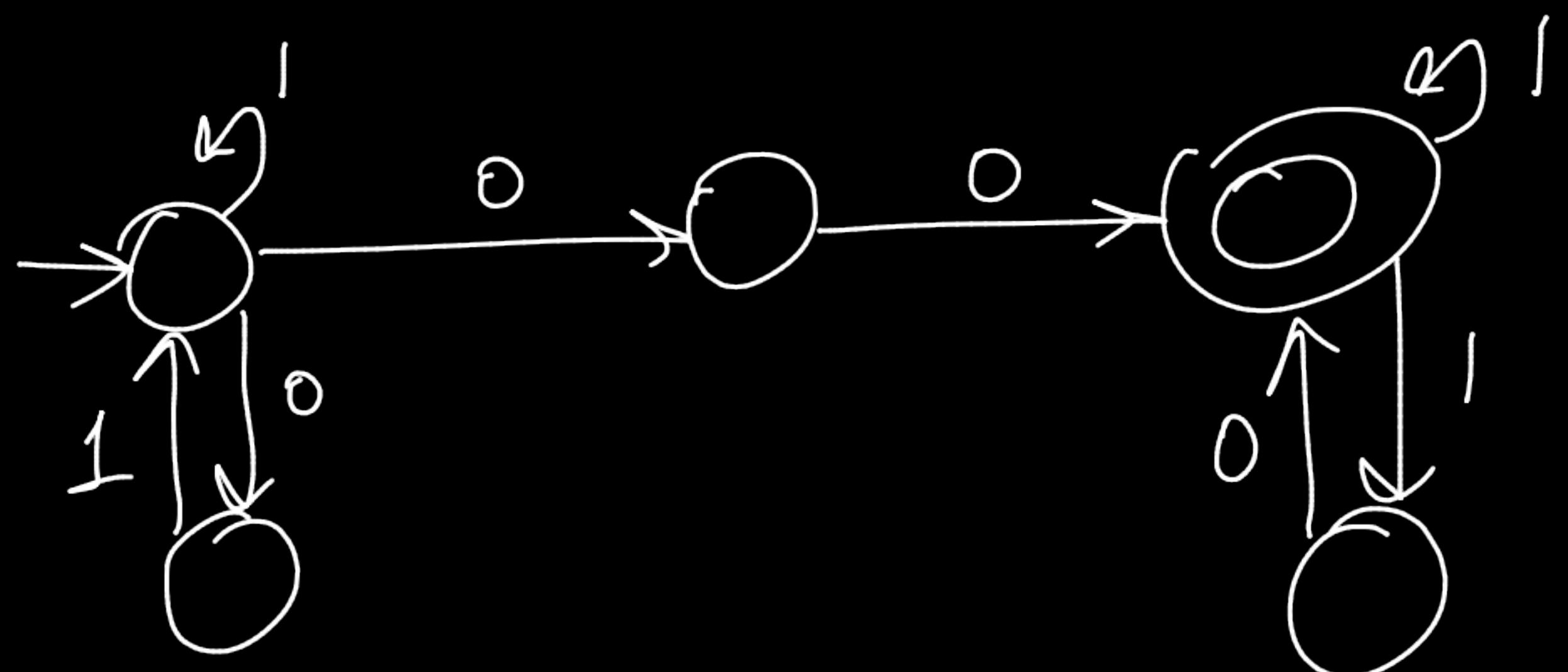


⑤

$$(1+01)^* \quad 00 \quad (1+10)^*$$

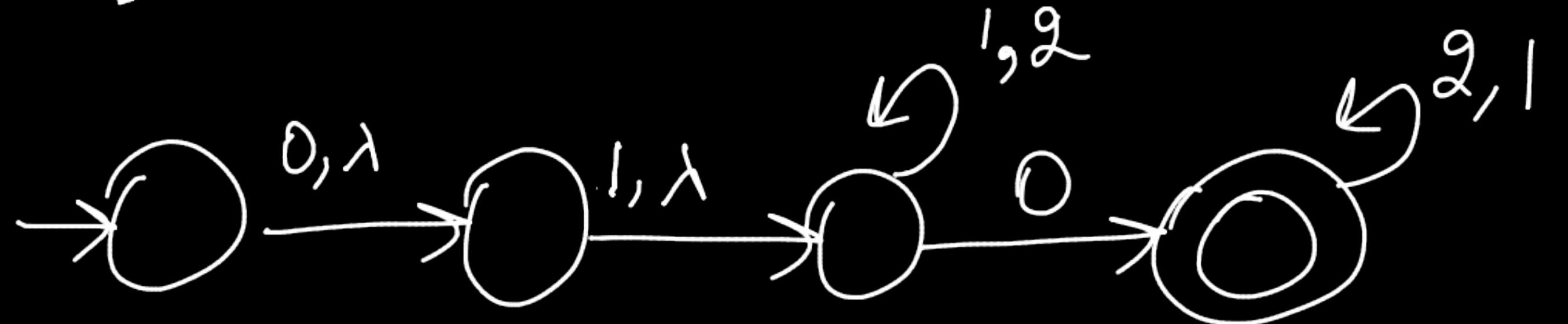


GTG (only
for
your
help)

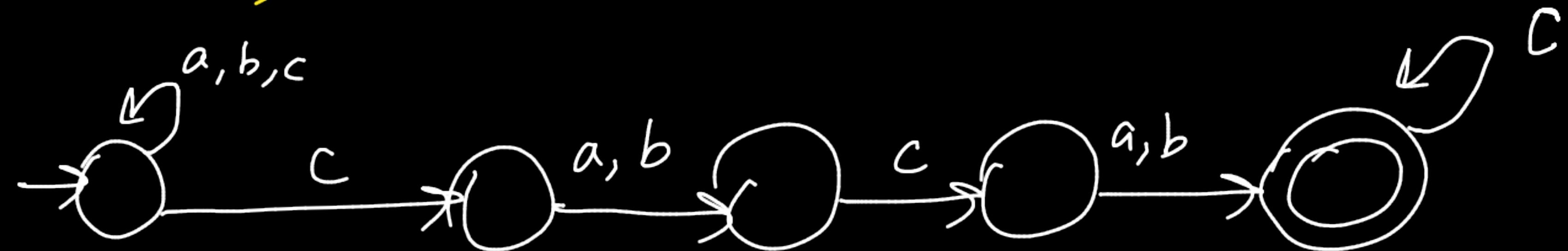


FA (solution)

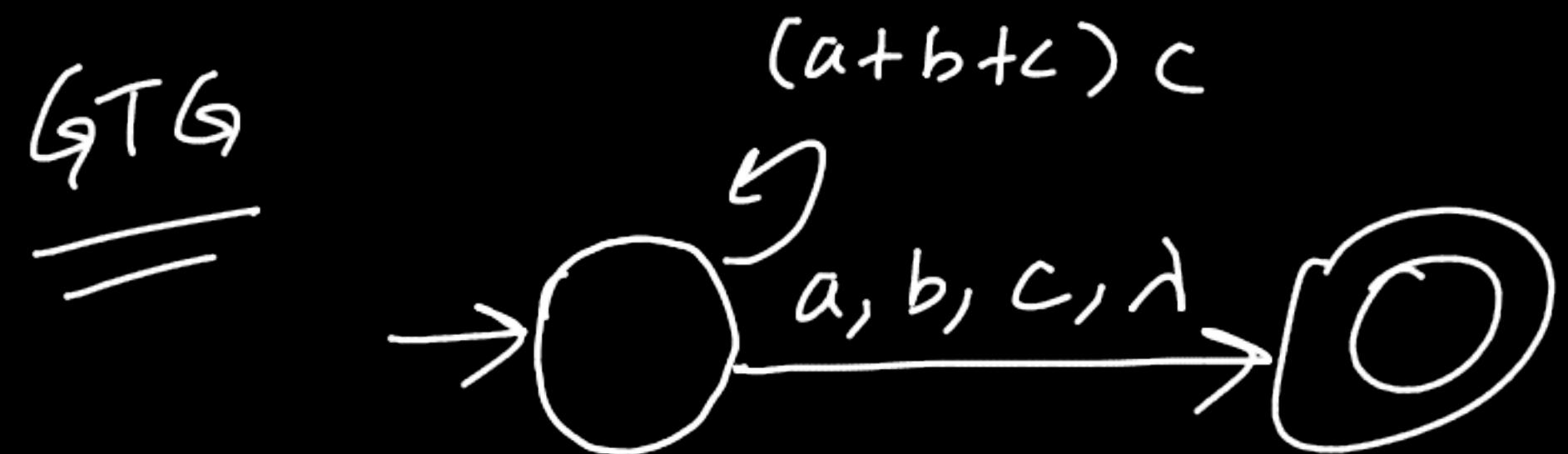
$$\textcircled{6} \quad (0+\lambda) \cdot (1+\lambda) \cdot (1+2)^* = 0 \cdot (2+1)^*$$



$$\textcircled{7} \quad (a+b+c)^* \subset (a+b) \subset (a+b)(c+\lambda)^*$$

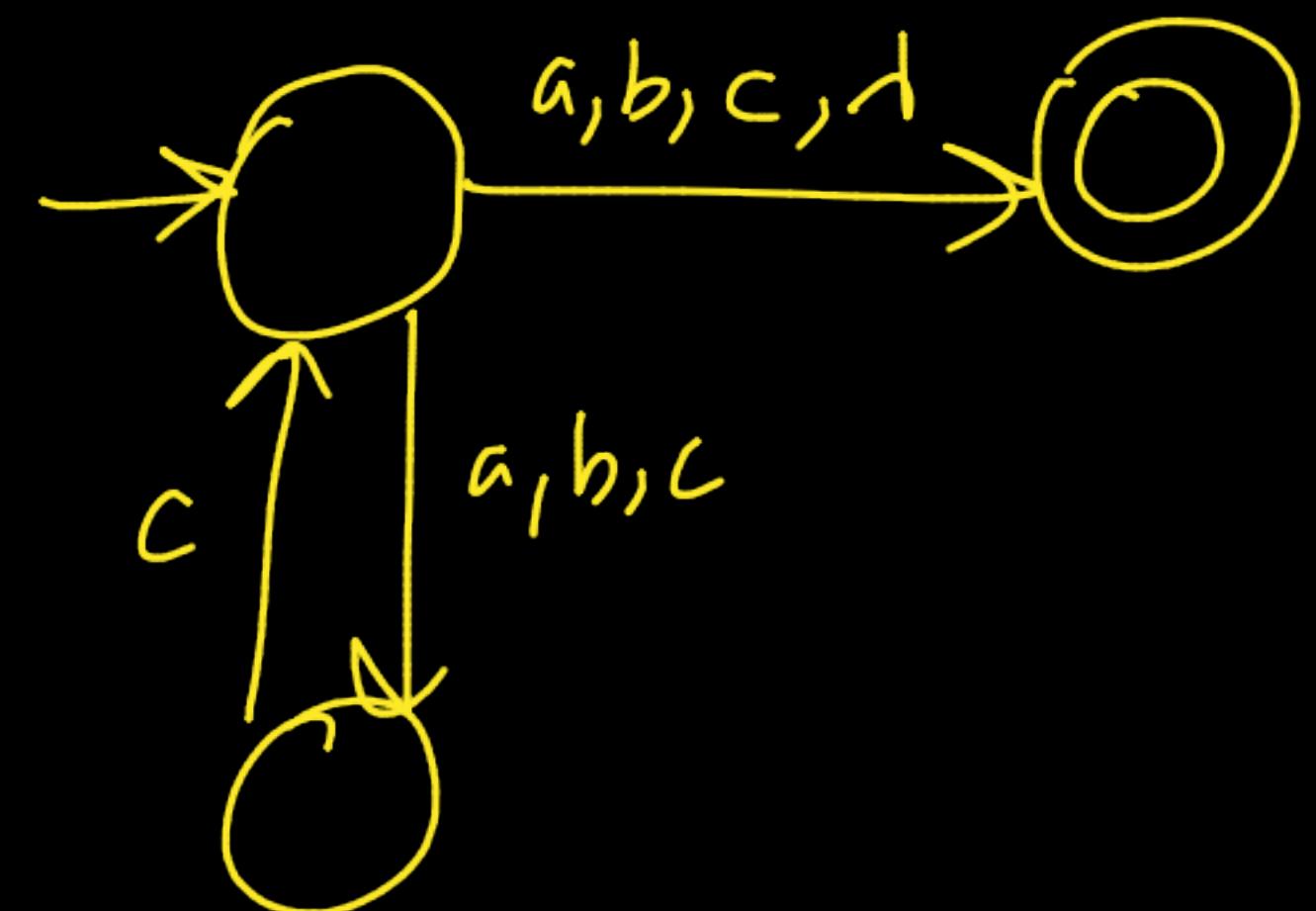


$$⑧ ((a+b+c)c)^* (a+b+c+\lambda)$$



} helps to decode what's in your
box

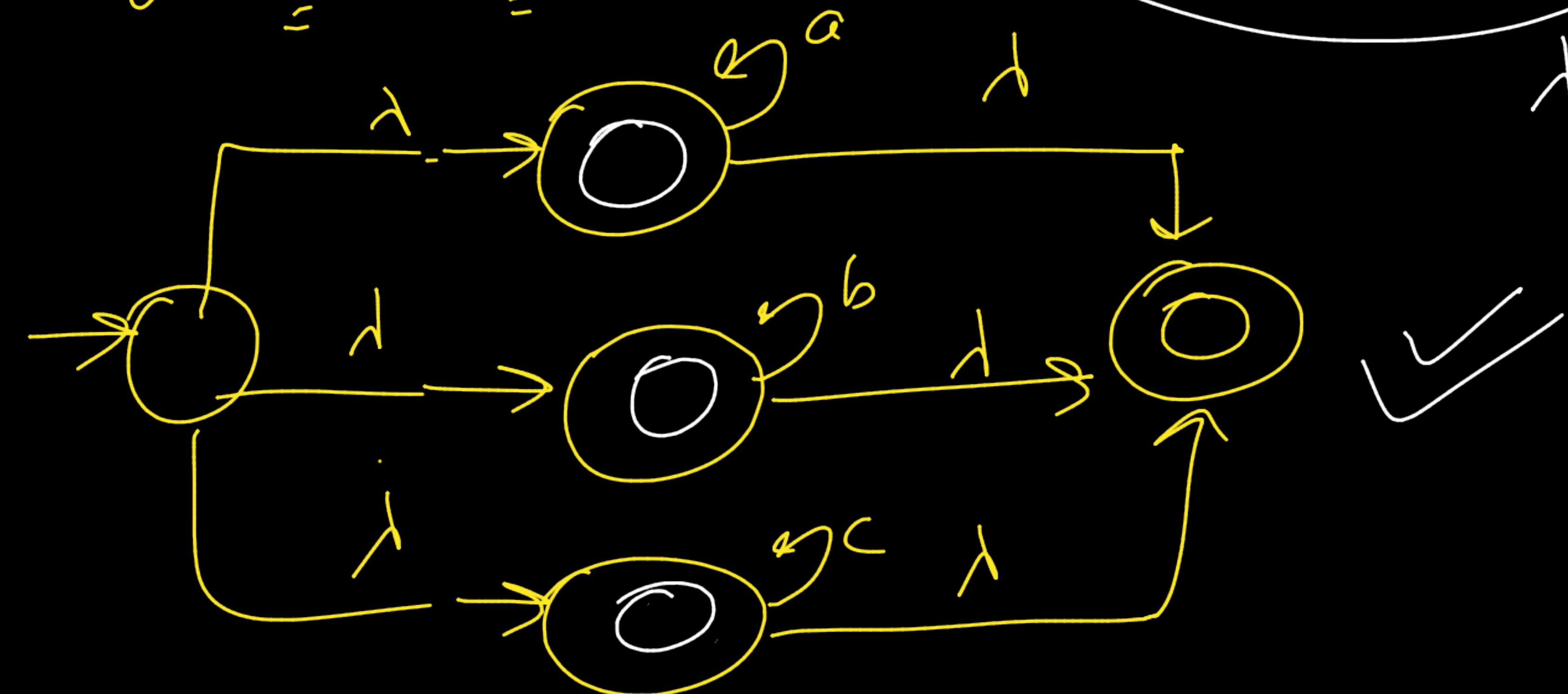
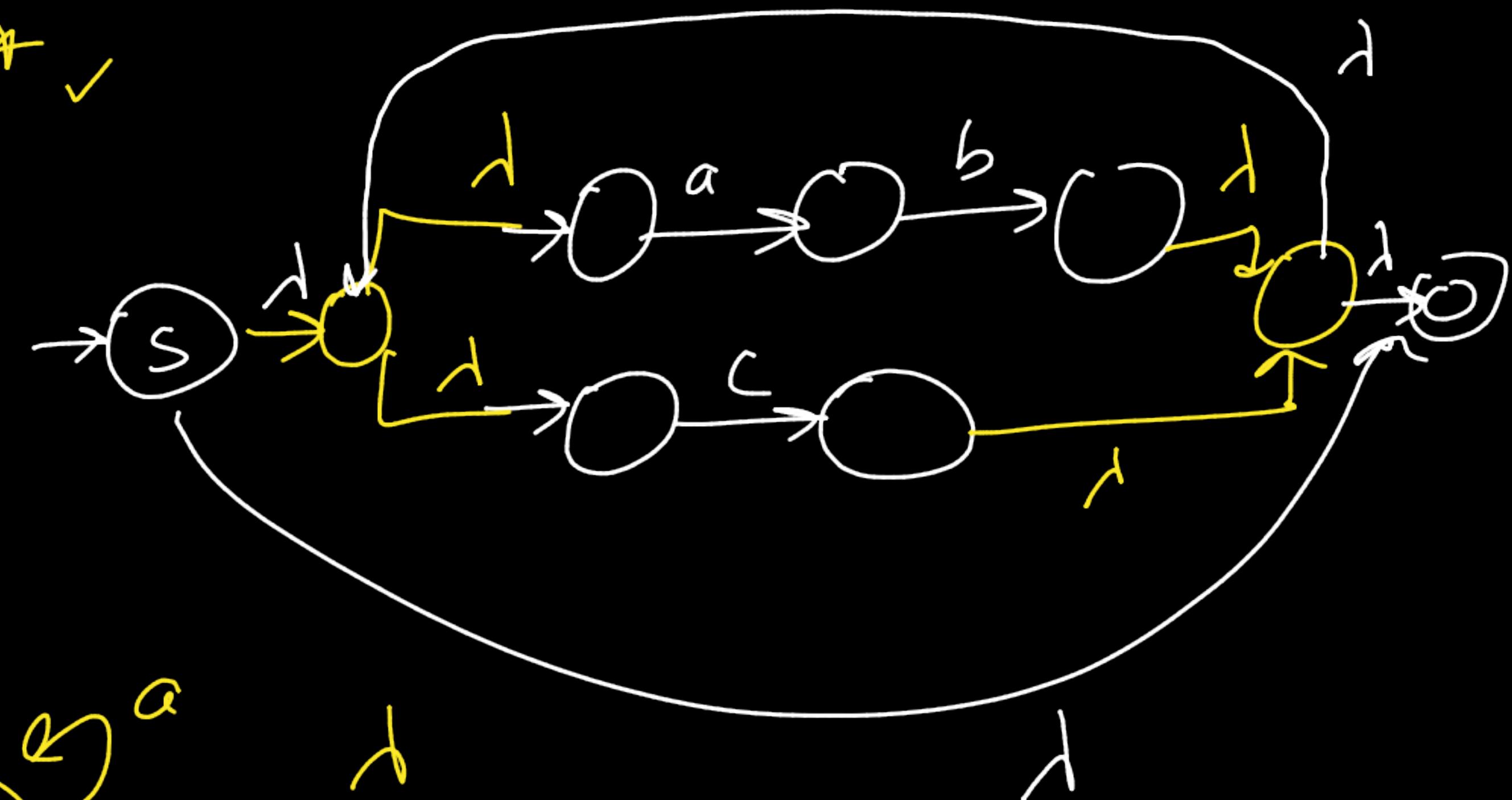
This is not the solution



9 $(a+b)^*$ $aa(a+b)^*$ ✓

10 $(ab+c)^*$ ✓

11 $R = a^* + b^* + c^*$



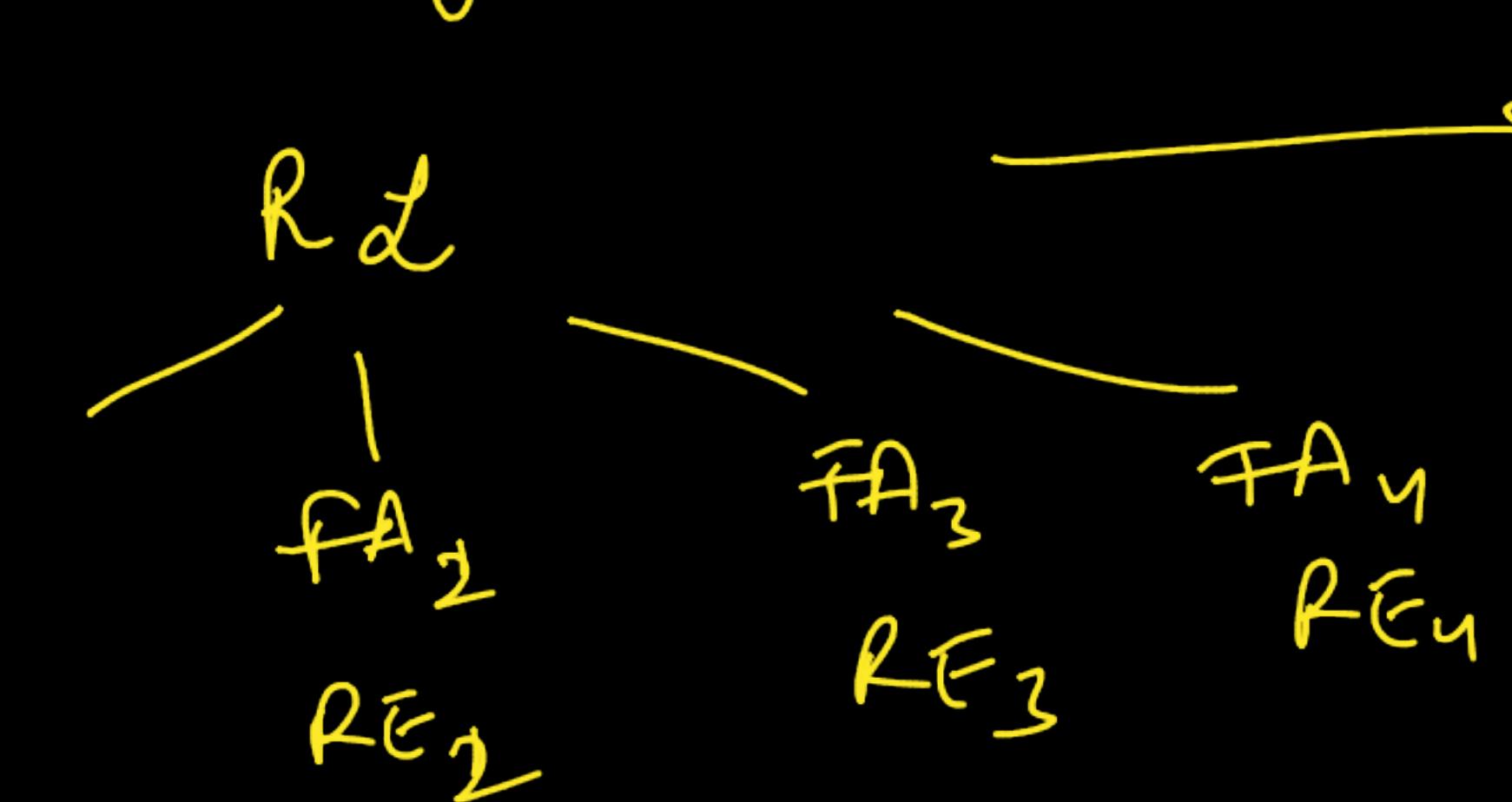
Possible variations

1) RE to FA

2) Given RE and FA Do they represent the same lang?
3) Given the lang pick w/c RE and FA accepts it

Multiple right answers.

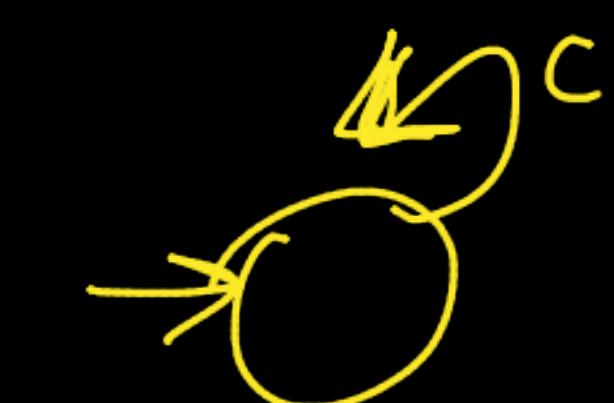
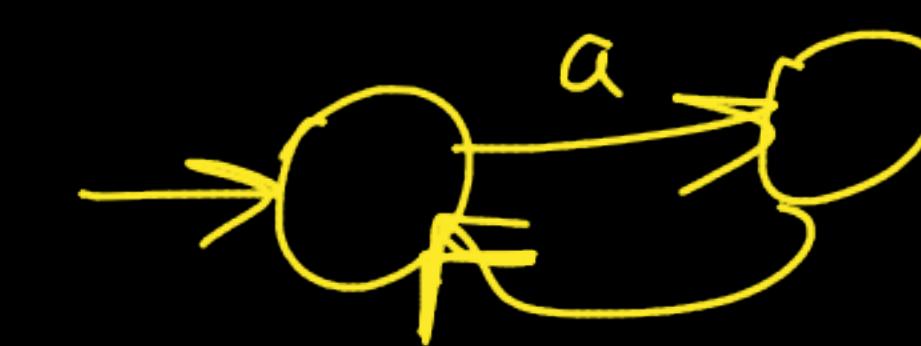
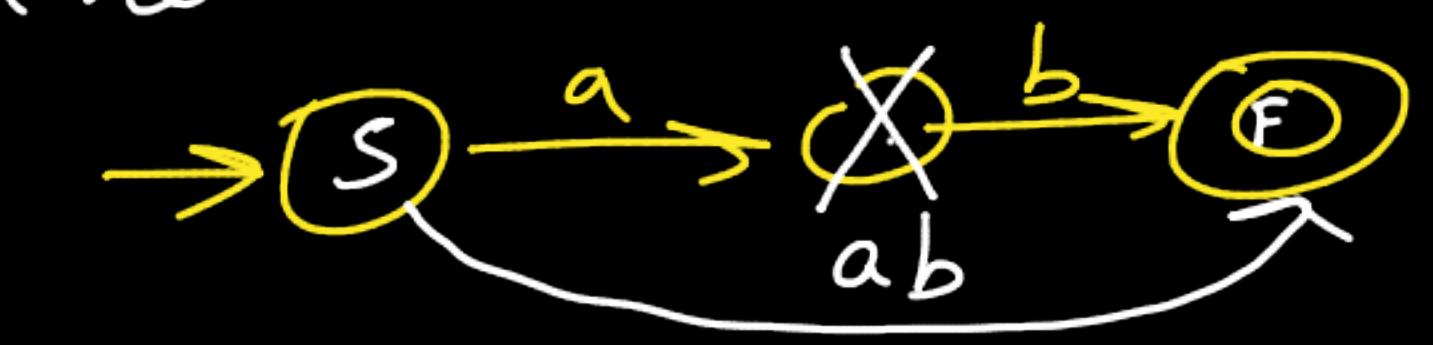
FA,
RE₁



minimal DFA
(unique)

Convert Finite Automata to RE (State Elimination Method)

Alg :-

- 1) There should be no incoming edge to your start state.
- 
- 
- You introduce a new start state with λ -transition to existing start state
- 2) No outgoing edge from the final state.
- Introduce a new final state with λ -trans.
- 
- 3) Start eliminating states
 - 4) $\rightarrow (S) \xrightarrow{R} (F)$

Q Which state should I eliminate first? \Rightarrow No

S Is there any specific order to be followed while eliminating states \Rightarrow No

You can delete other states in any order



Regex

\Rightarrow will look diff for diff orders picked up



B A

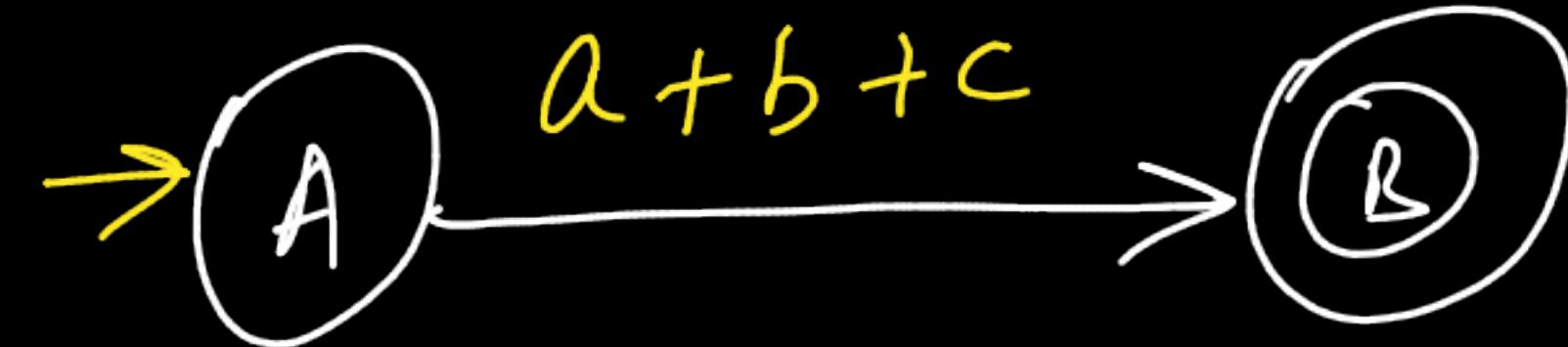
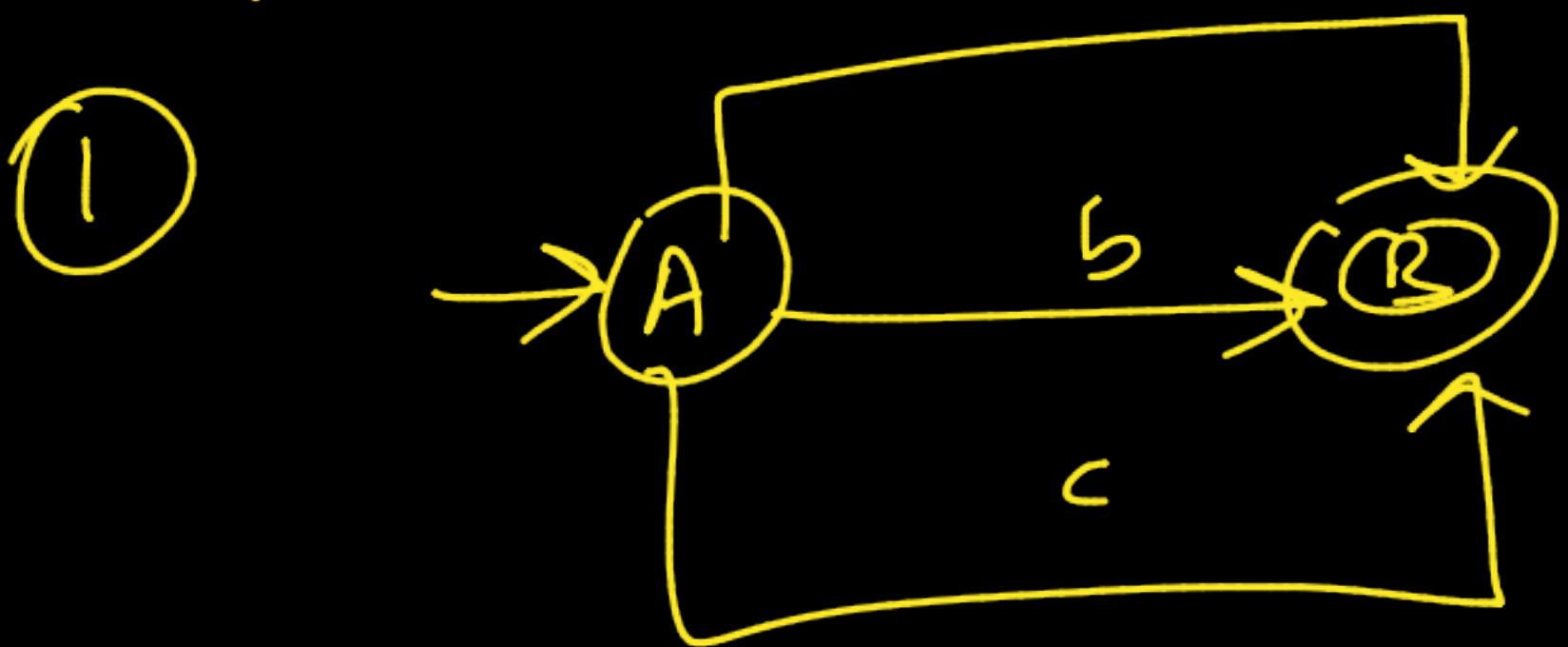
D Q

A C

C D

Examples to convert a FA to RE

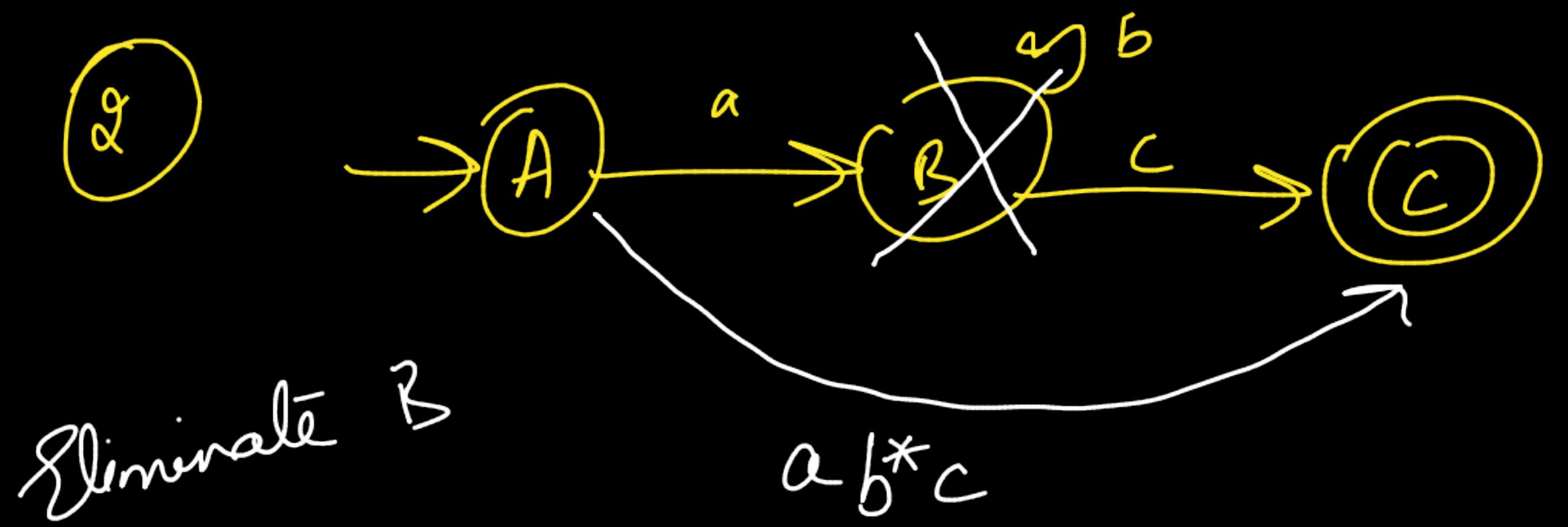
①



$$RE = (a+b+c)$$

Checks

- Any incoming edge to your start state \rightarrow No
- Any outgoing edge from final state \rightarrow No
- whatever is left of start & final state that's your regex

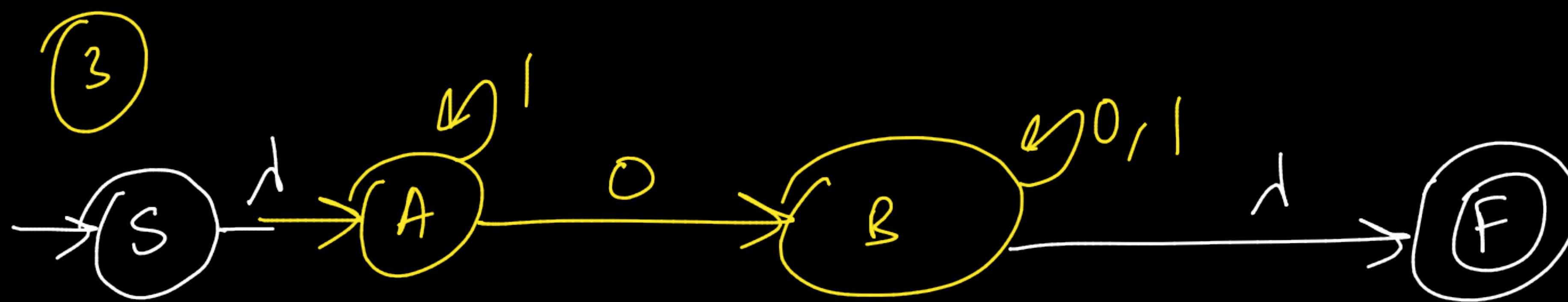


éliminale B

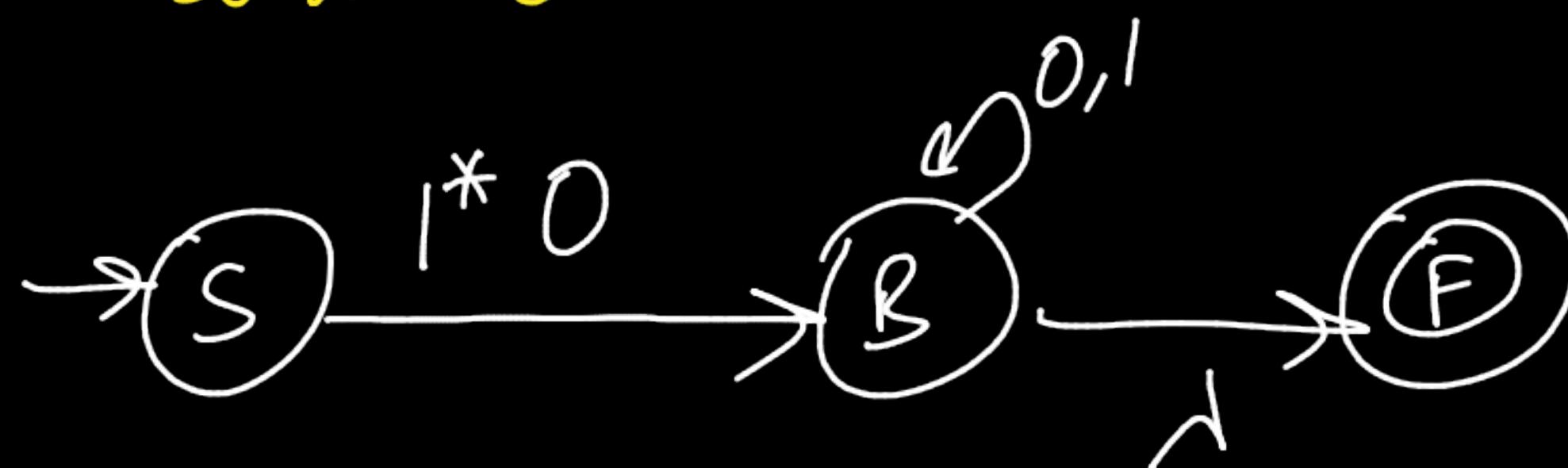
$a b^* c$



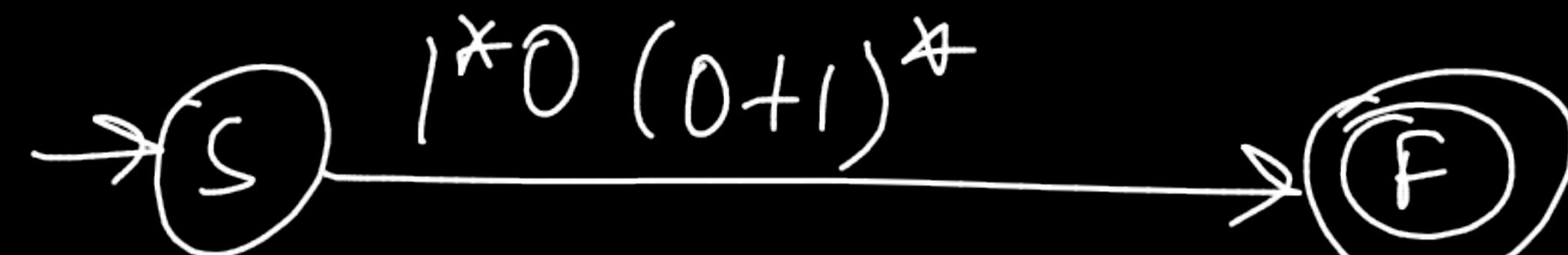
③



Eliminate A



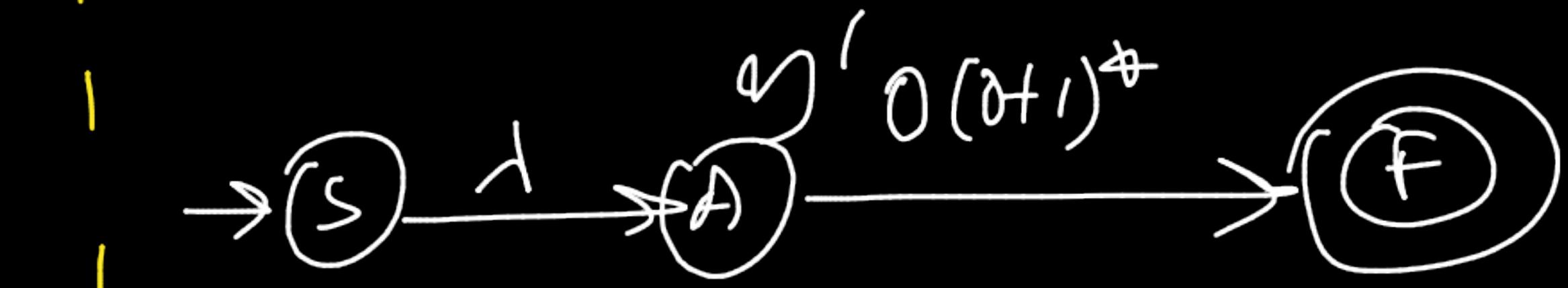
Eliminate B



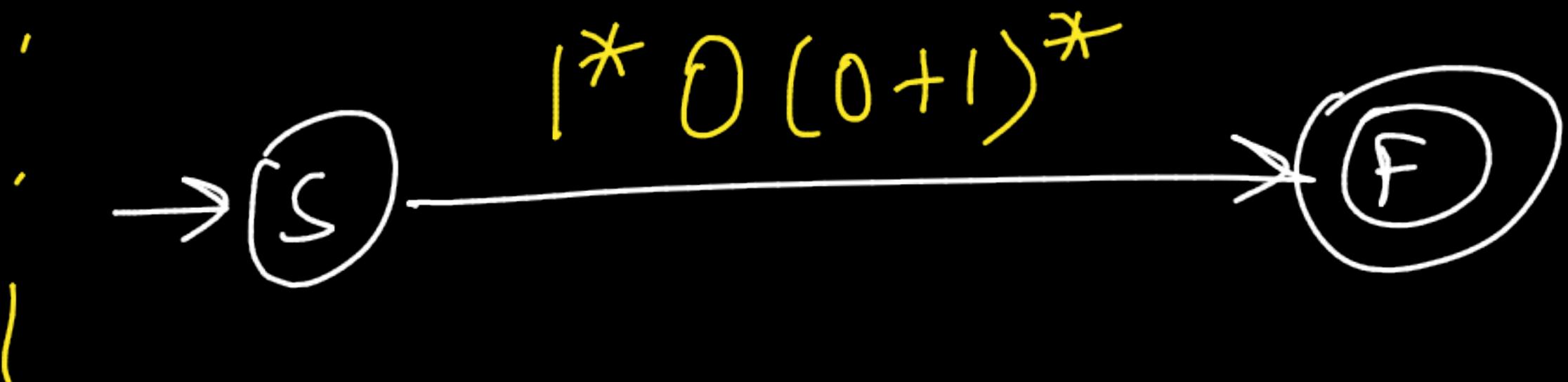
Same

$$\lambda \cdot w = w$$
$$w \cdot \lambda = w$$

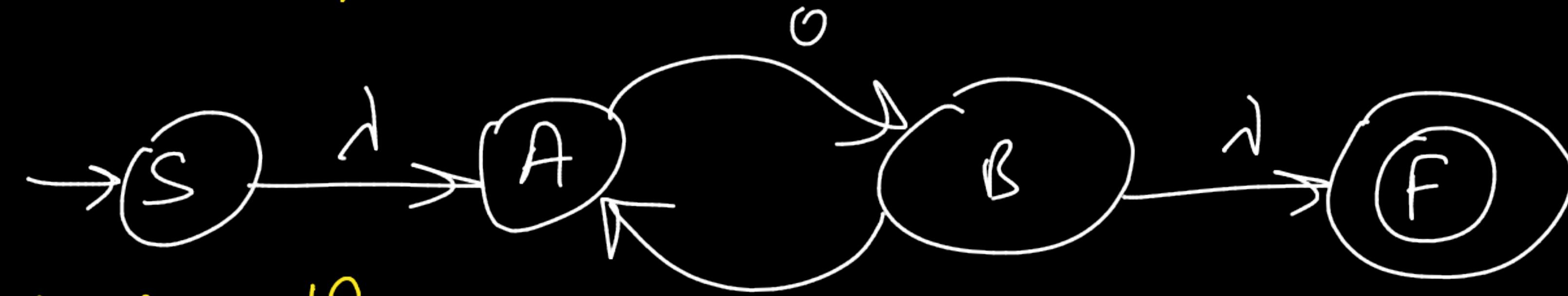
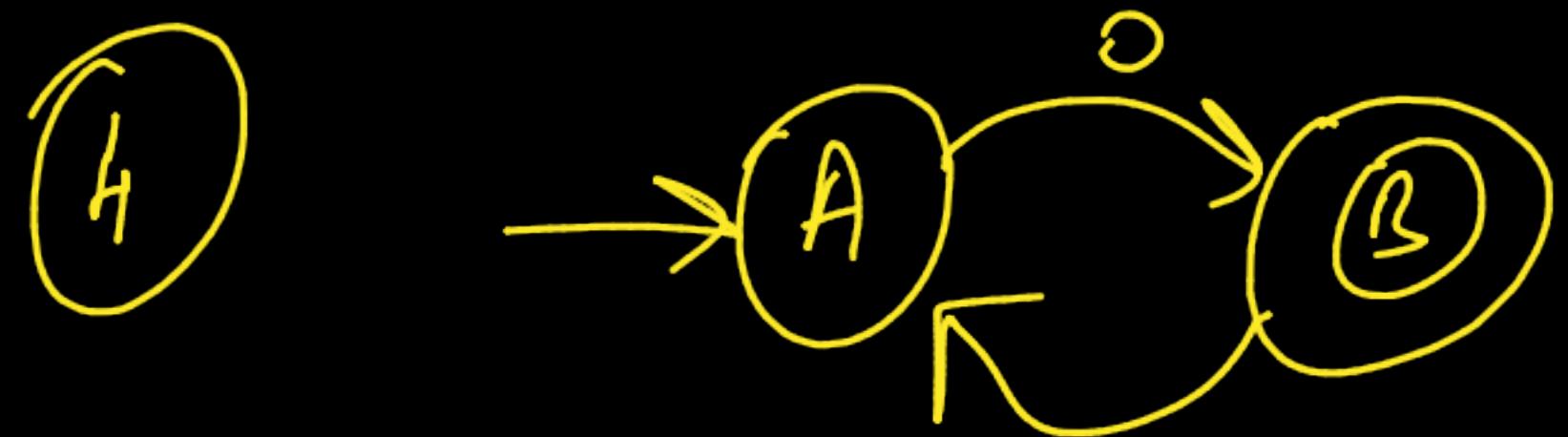
Eliminate B



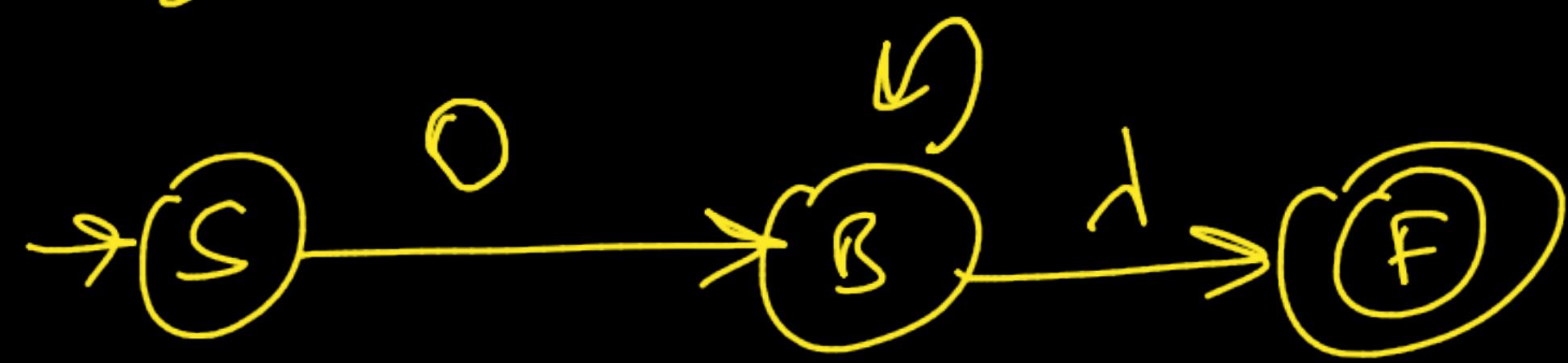
Eliminate A



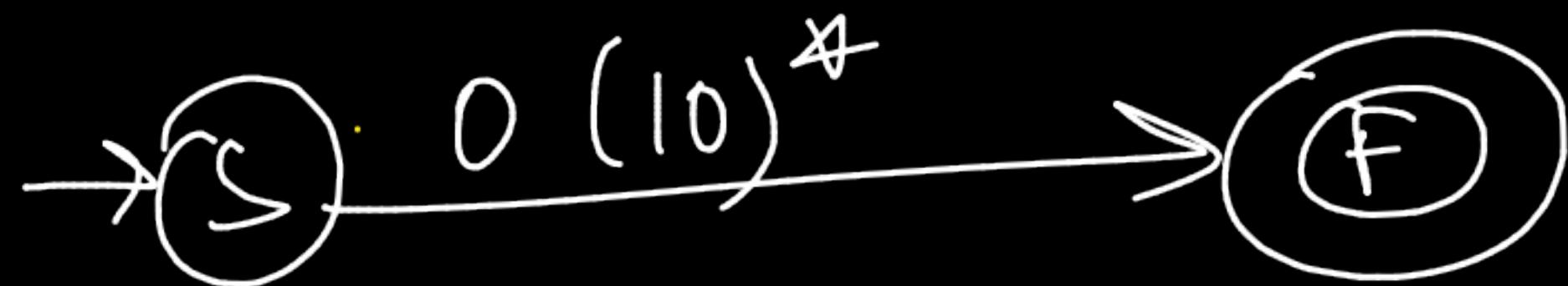
answer



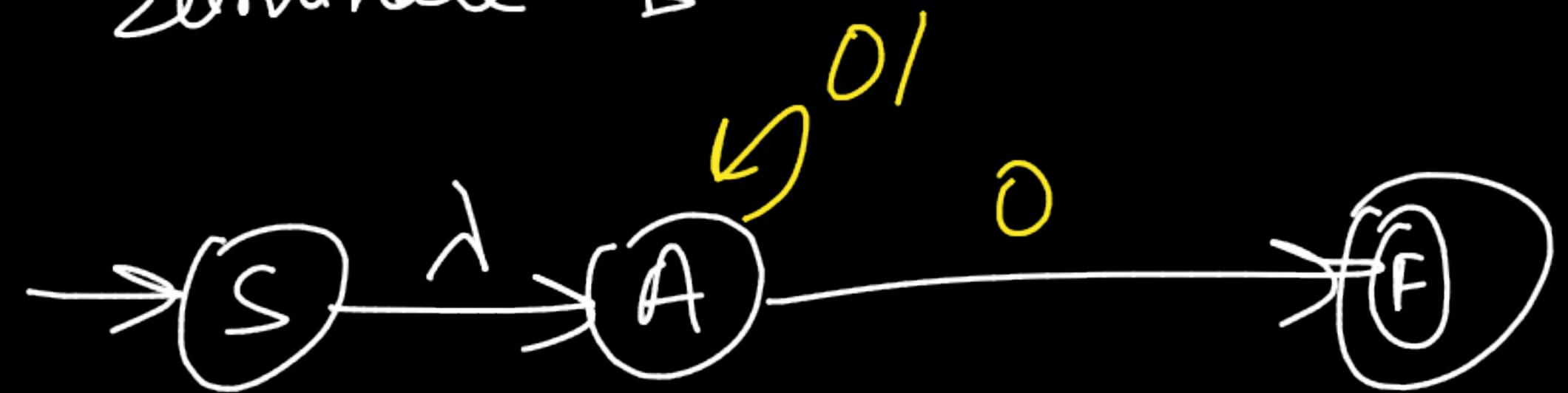
Eliminate A



Eliminate B



Eliminate B



Eliminate A

