

Science informatique

CM8

Antonio E. Porreca

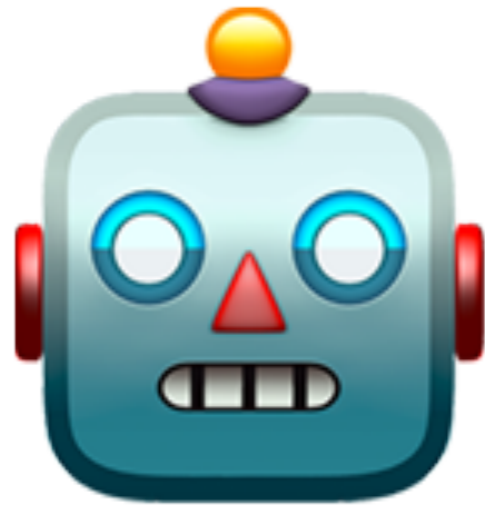
aeporreca.org/scienceinfo

Exercice 3 du TD6

Recherche de mots dans un texte



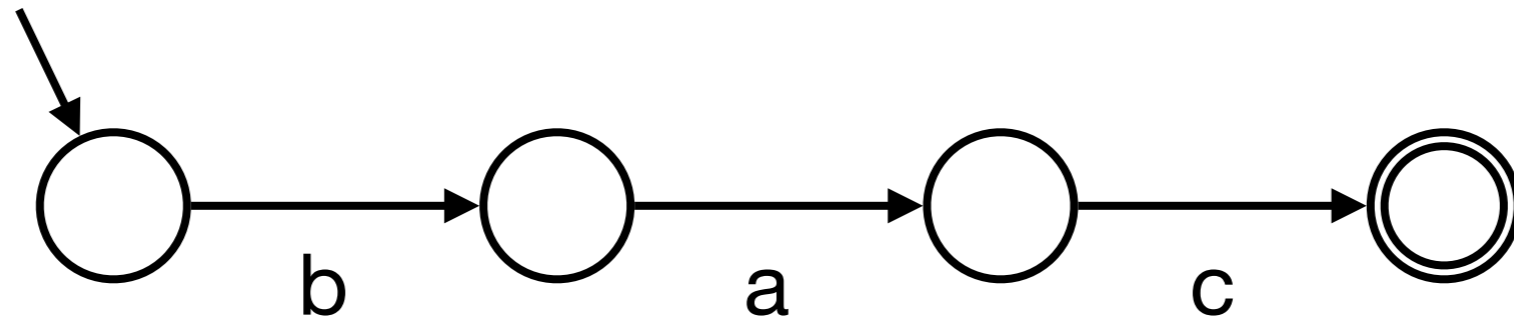
Recherche de mots dans un texte



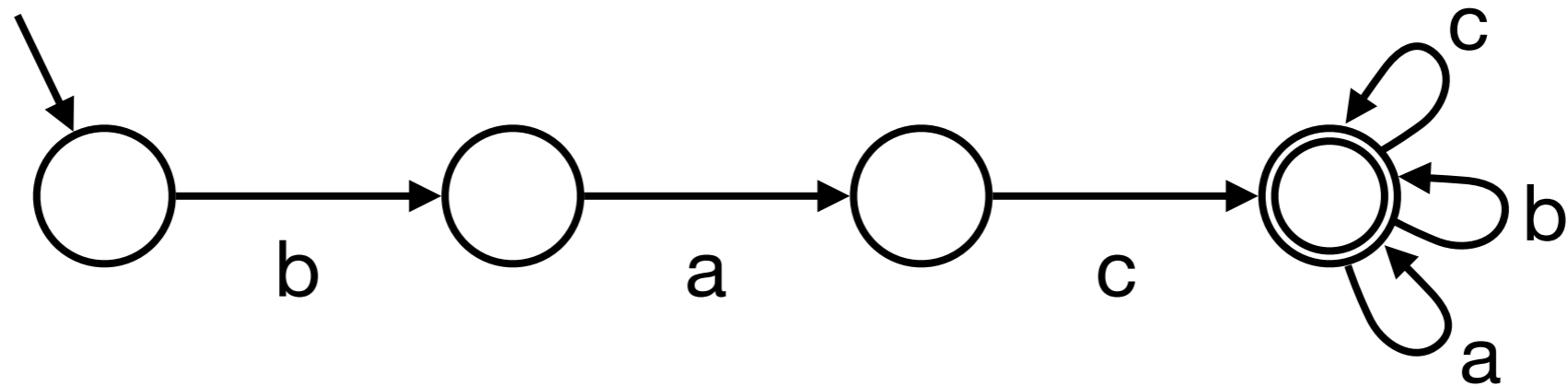
Exemple de recherche

- On utilise, pour simplicité, l'alphabet réduit {a, b, c}
- On veut décrire un automate qui accepte un texte si et seulement si il contient le mot « bac »
- Textes acceptés : ab**ac**aa, ba**cc**cb, bcc**ba**c
- Textes non acceptés : abcaaa, cab

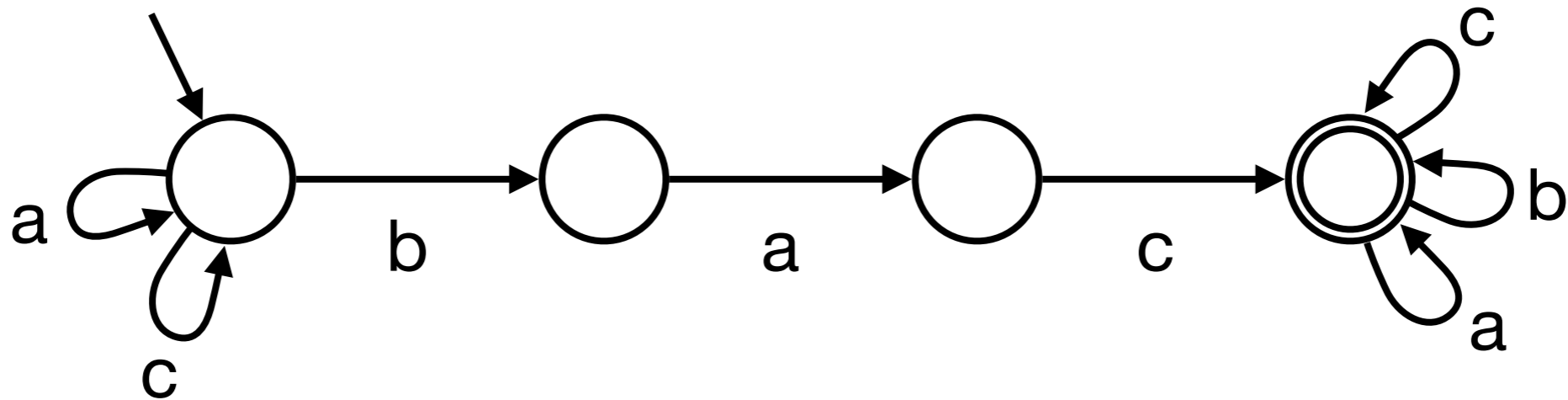
Automate pour le textes qui contient « bac »



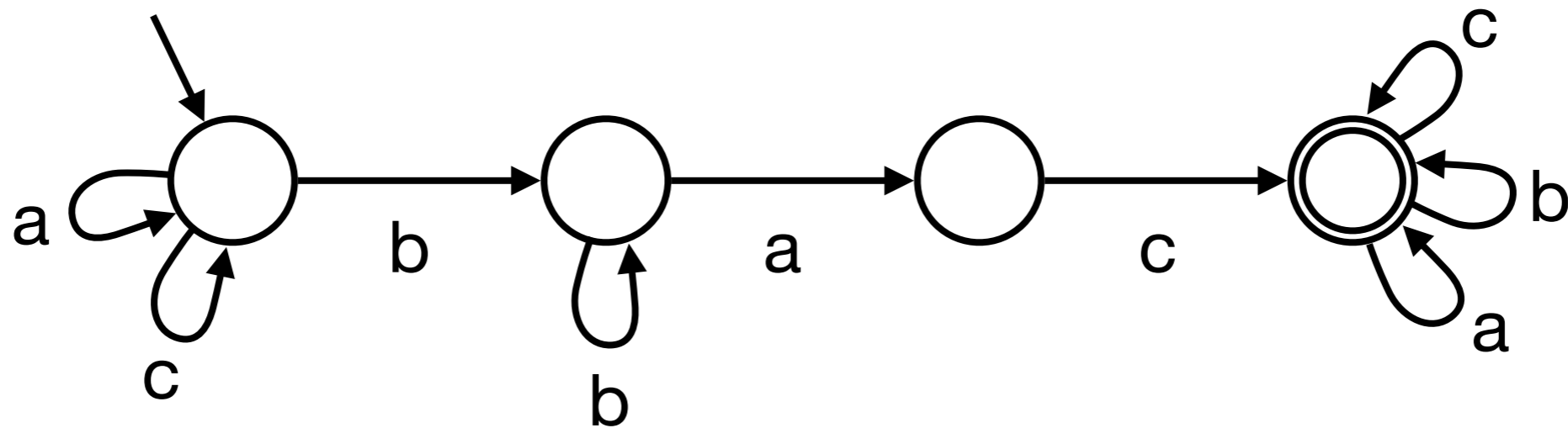
Automate pour le textes qui contient « bac »



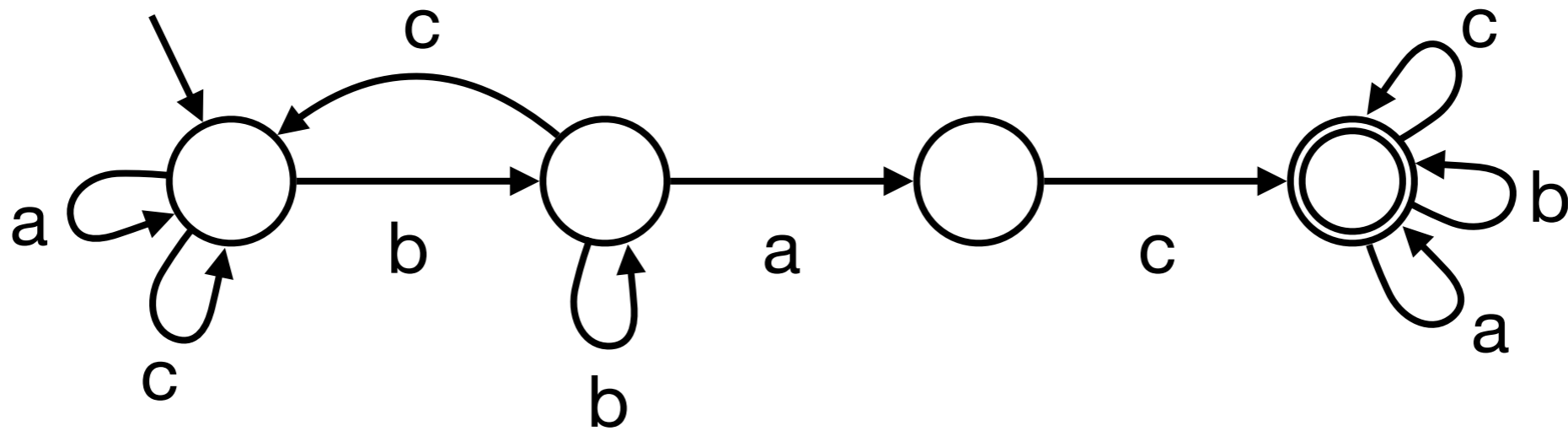
Automate pour le textes qui contient « bac »



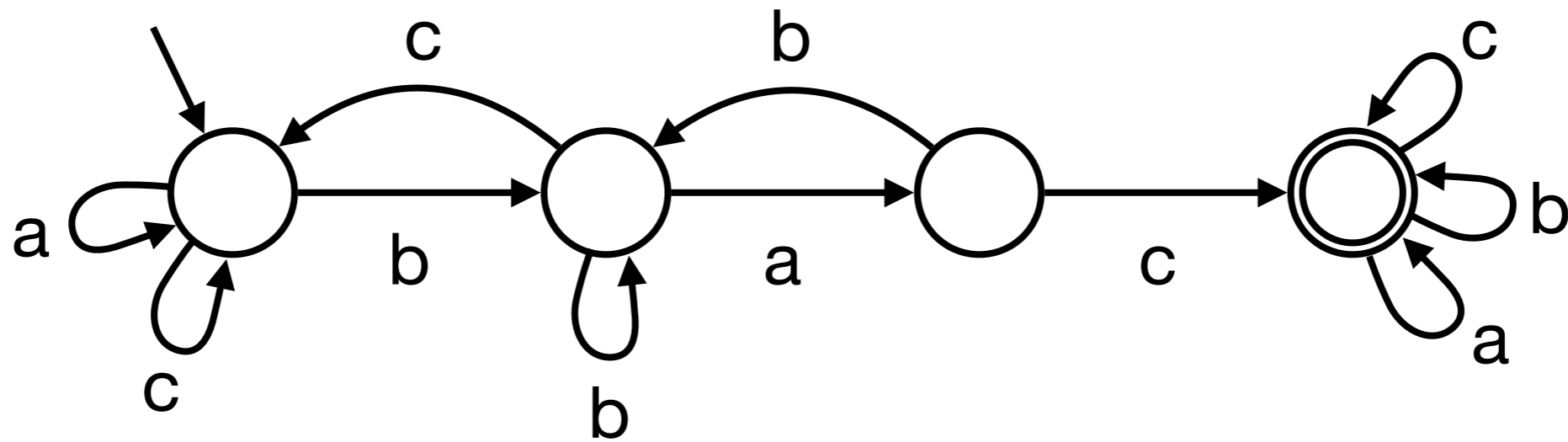
Automate pour le textes qui contient « bac »



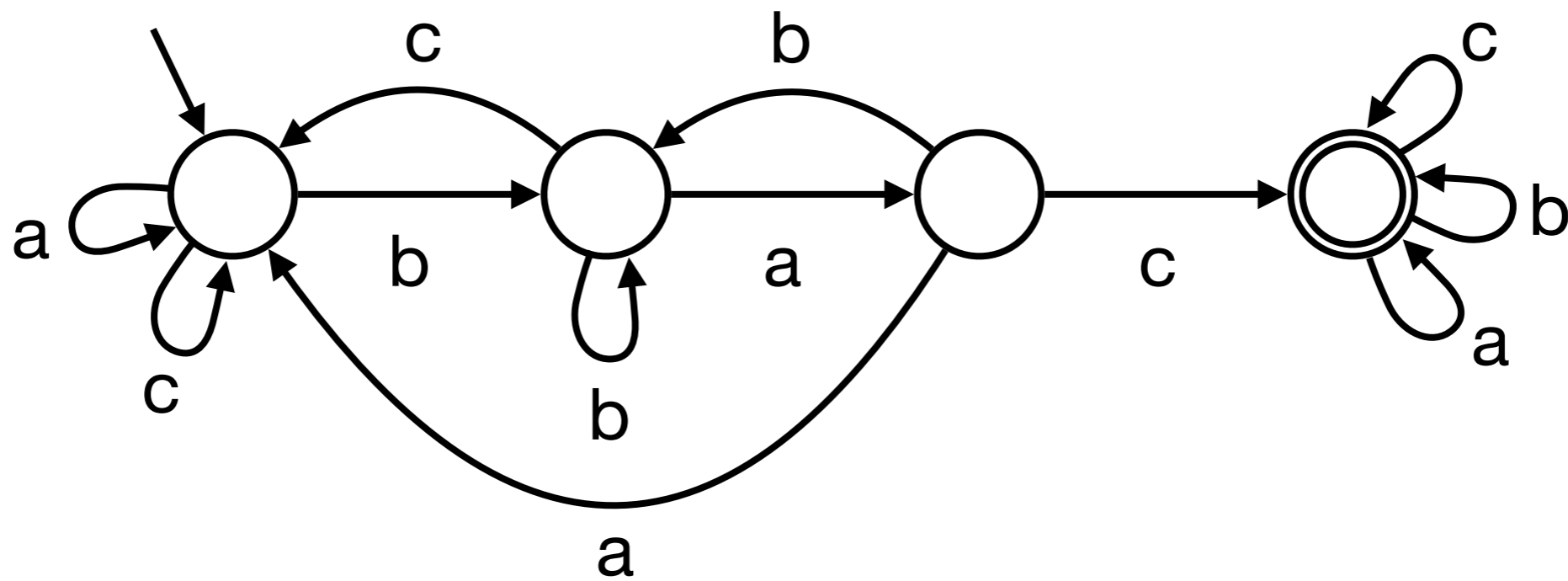
Automate pour le textes qui contient « bac »



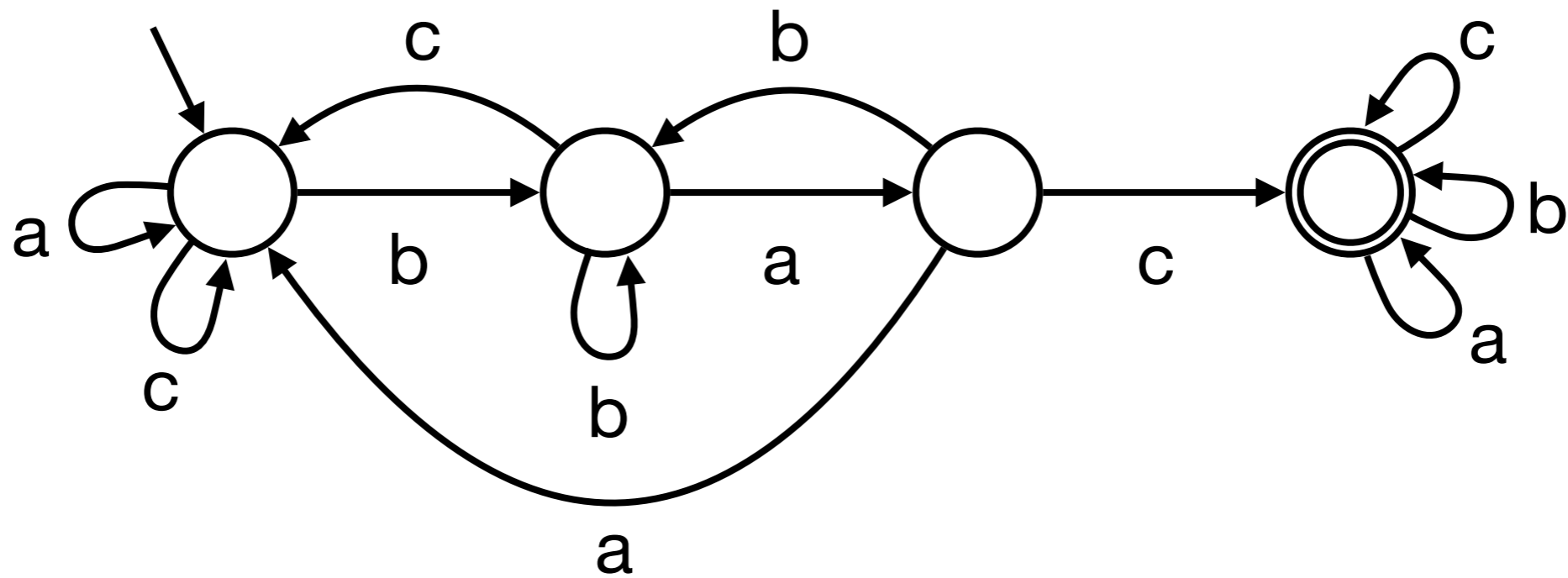
Automate pour le textes qui contient « bac »



Automate pour le textes qui contient « bac »

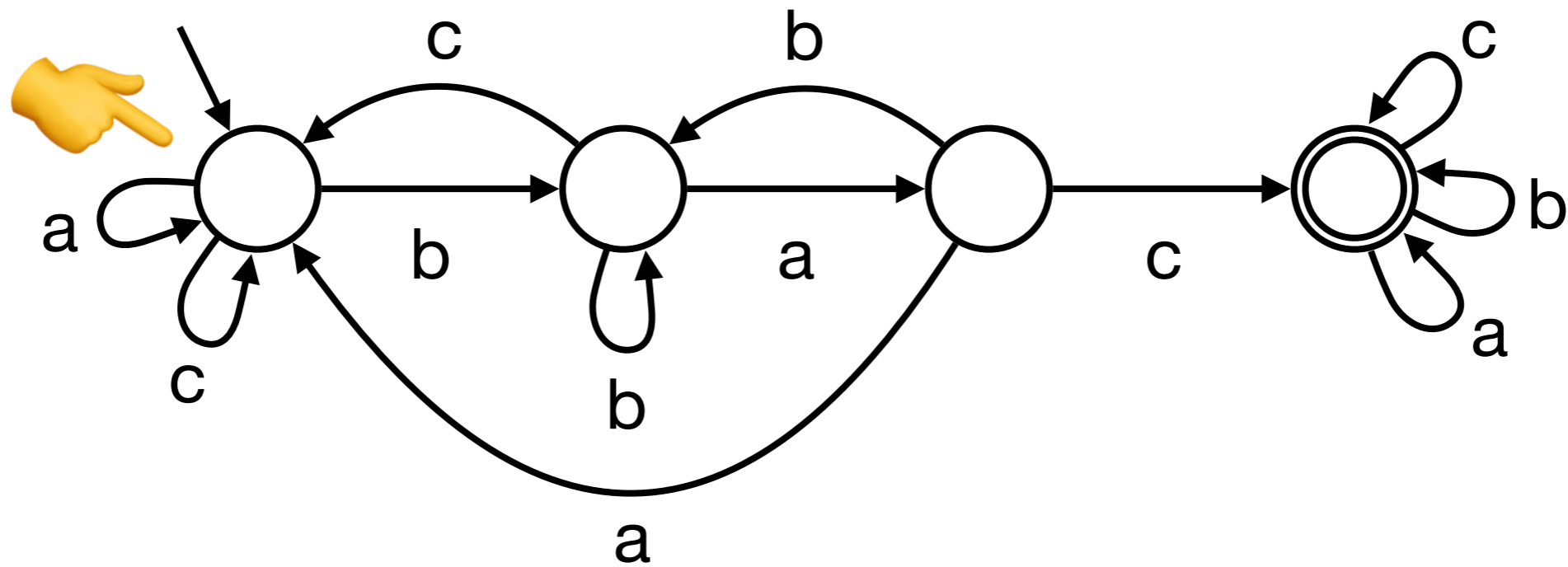


Exemples de recherche



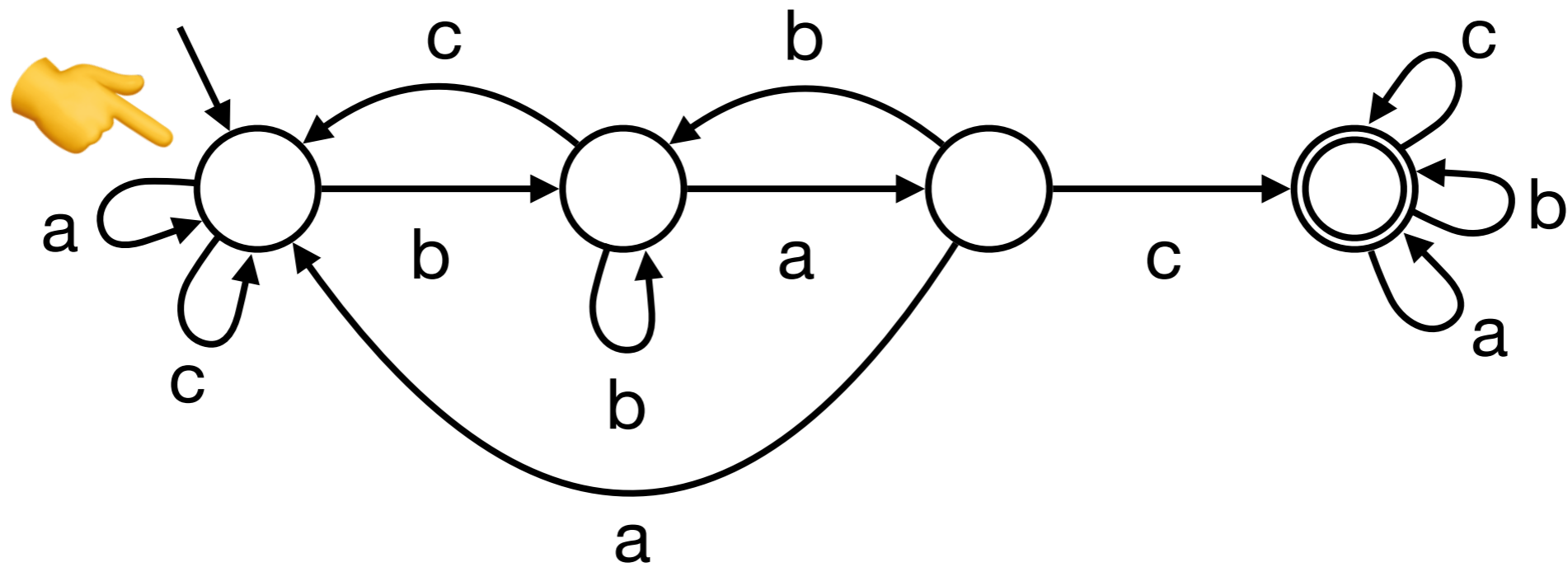
abaca~~aa~~

Exemples de recherche



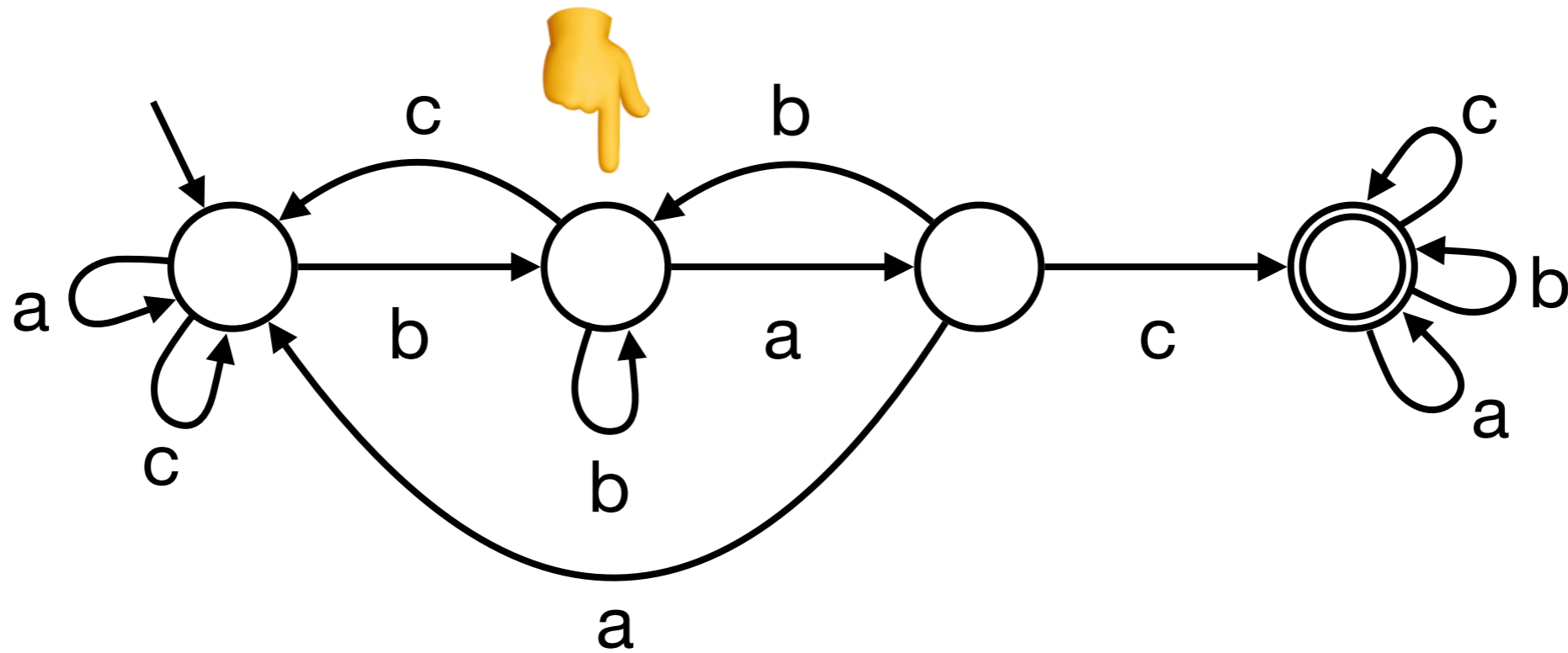
abaca~~aa~~

Exemples de recherche



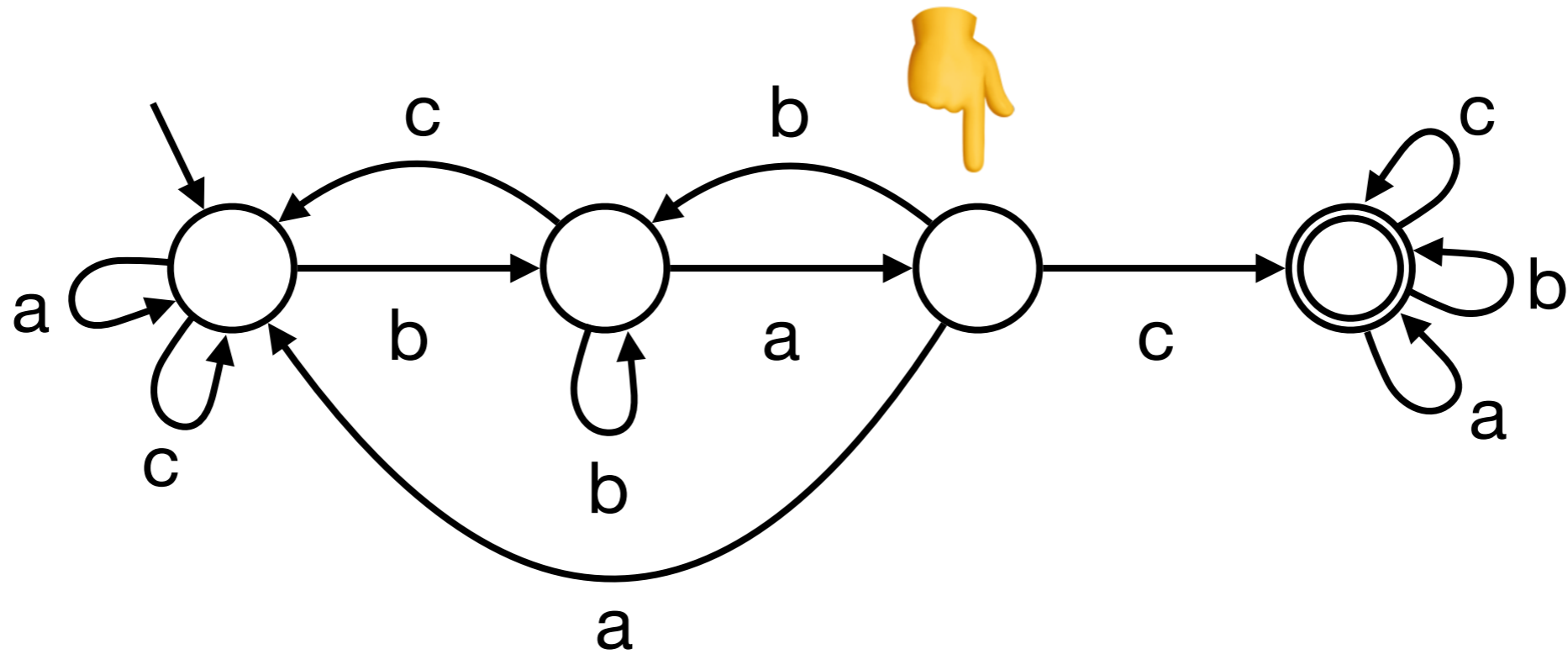
abacaaa

Exemples de recherche



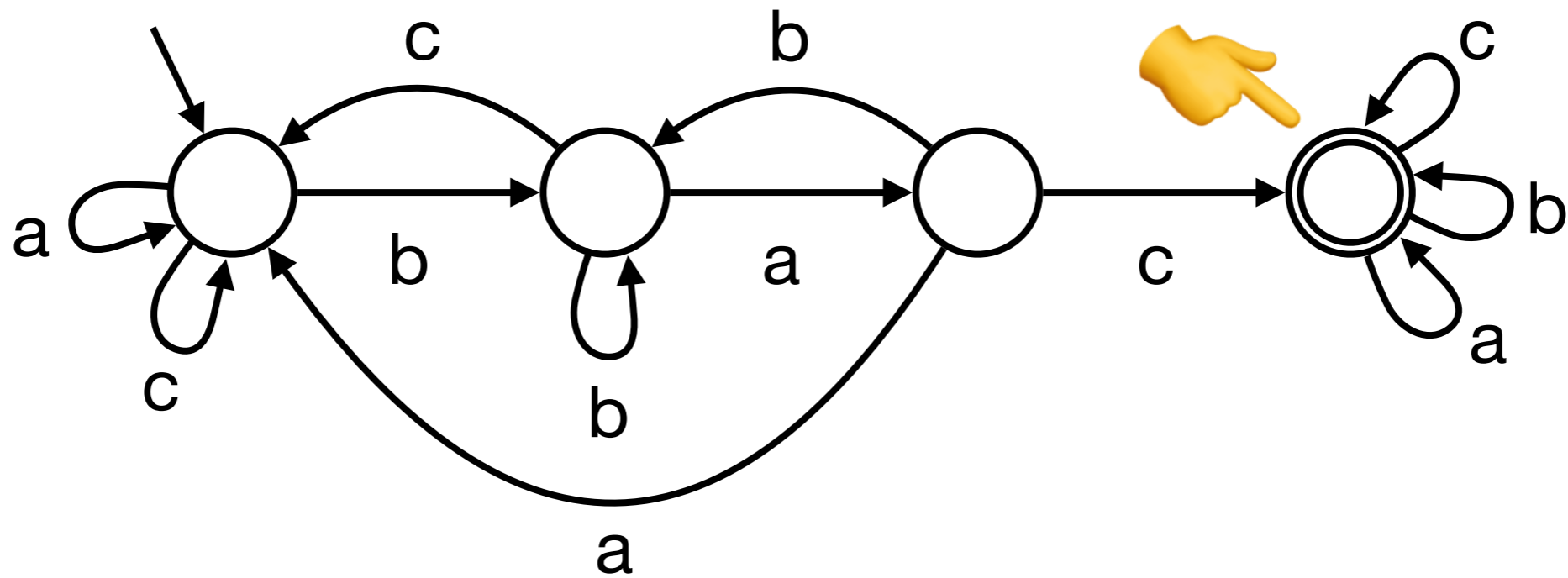
abaca~~aa~~

Exemples de recherche



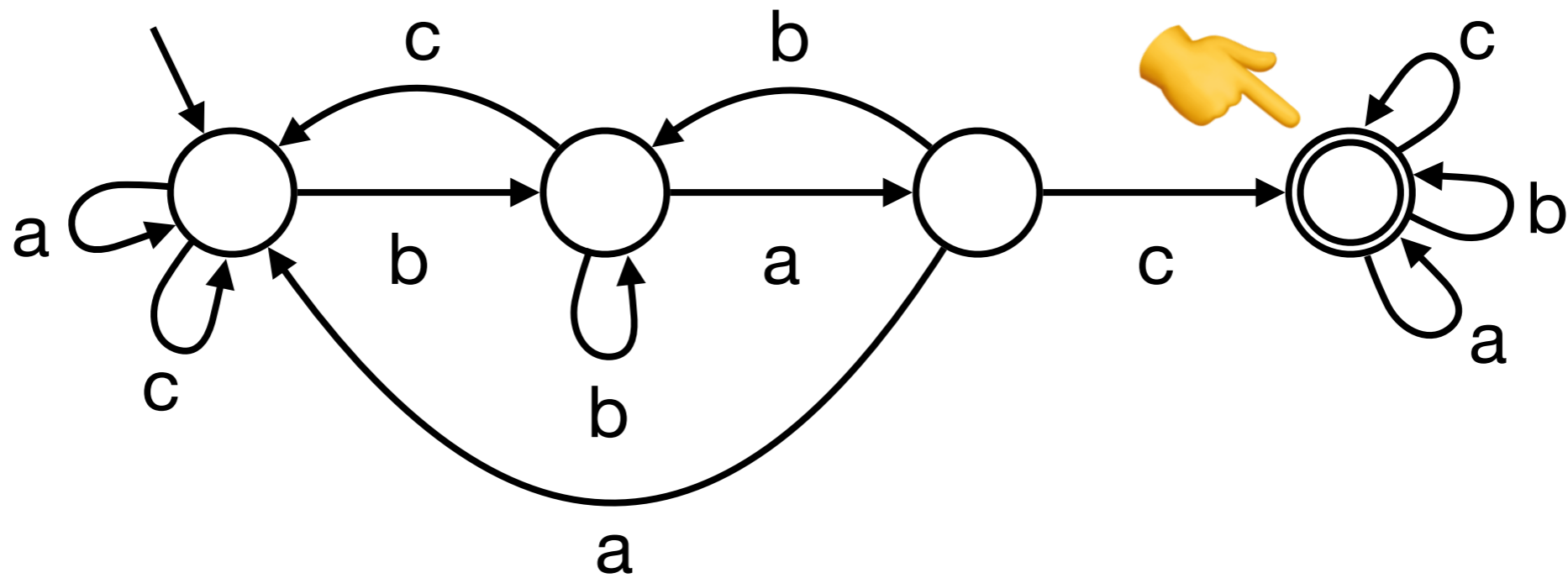
abacaaa

Exemples de recherche



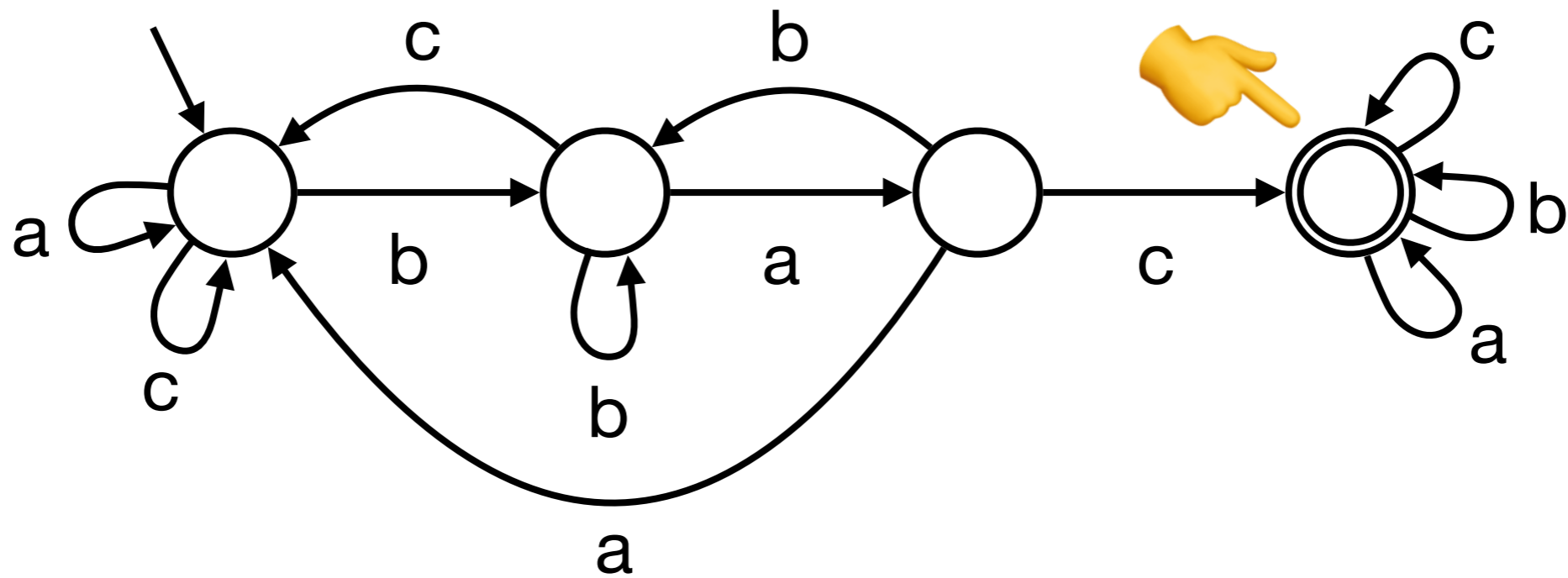
abacaaa

Exemples de recherche



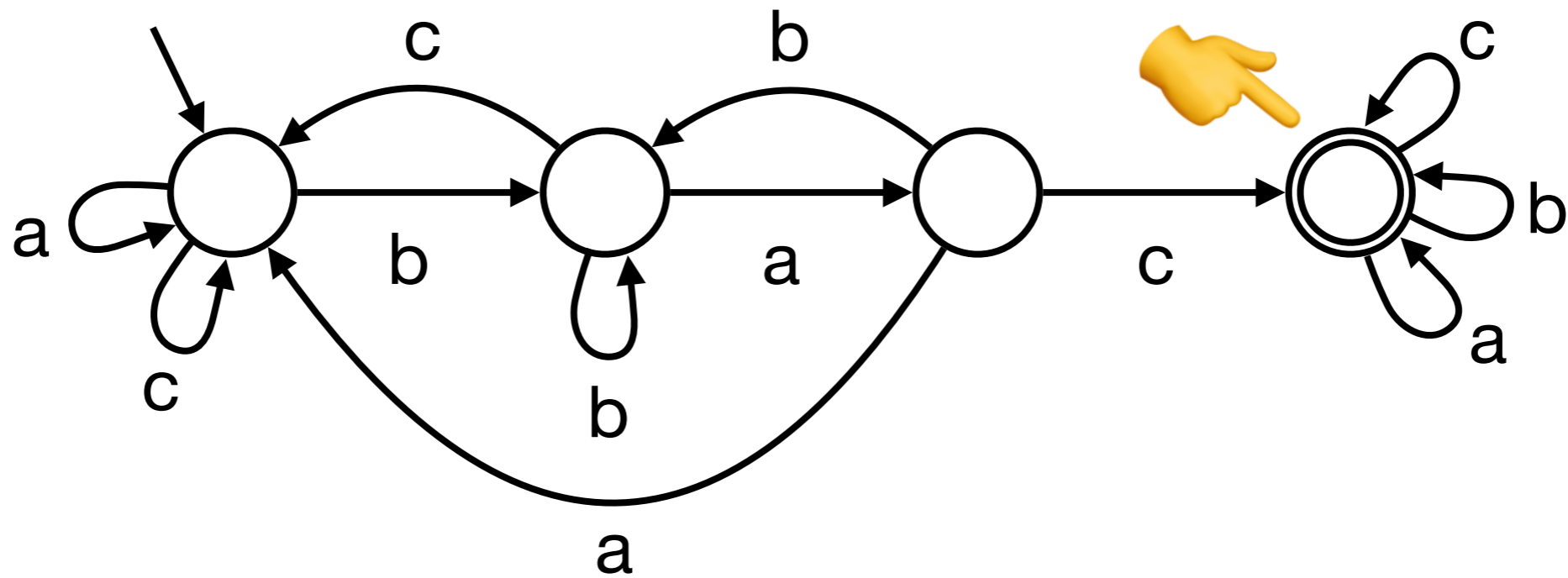
abaca~~aa~~

Exemples de recherche



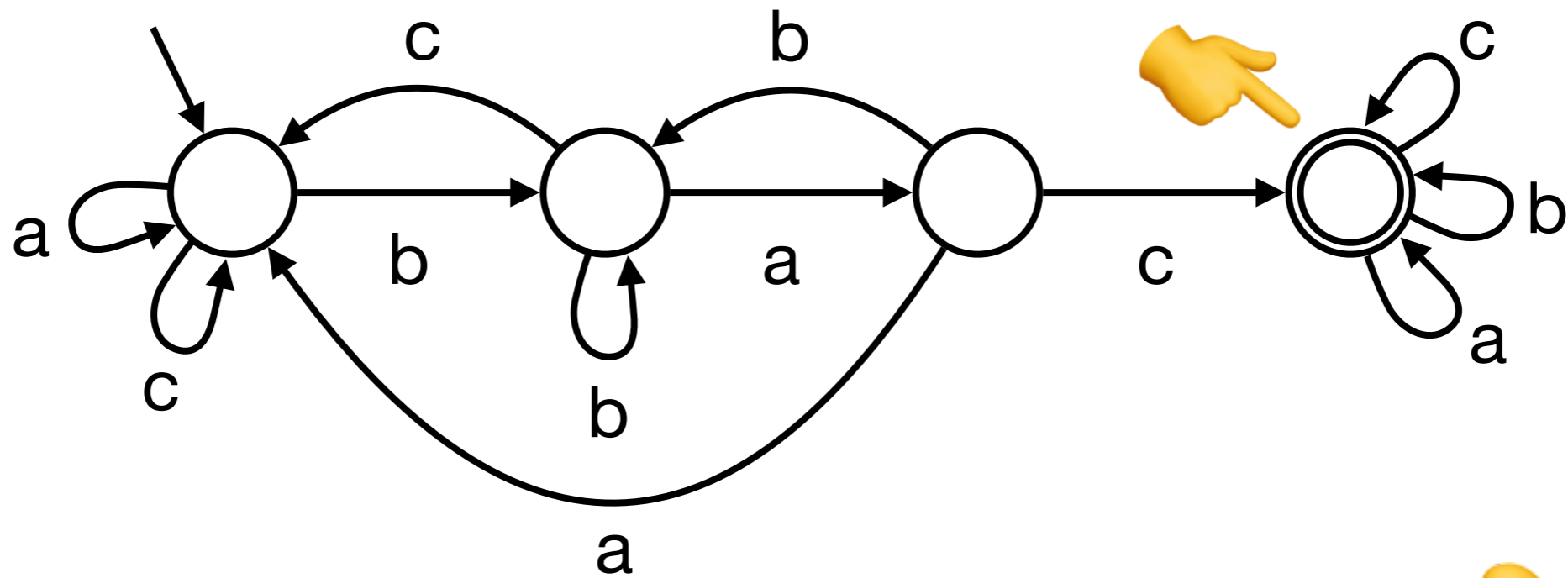
abacaaa

Exemples de recherche



abacaaa

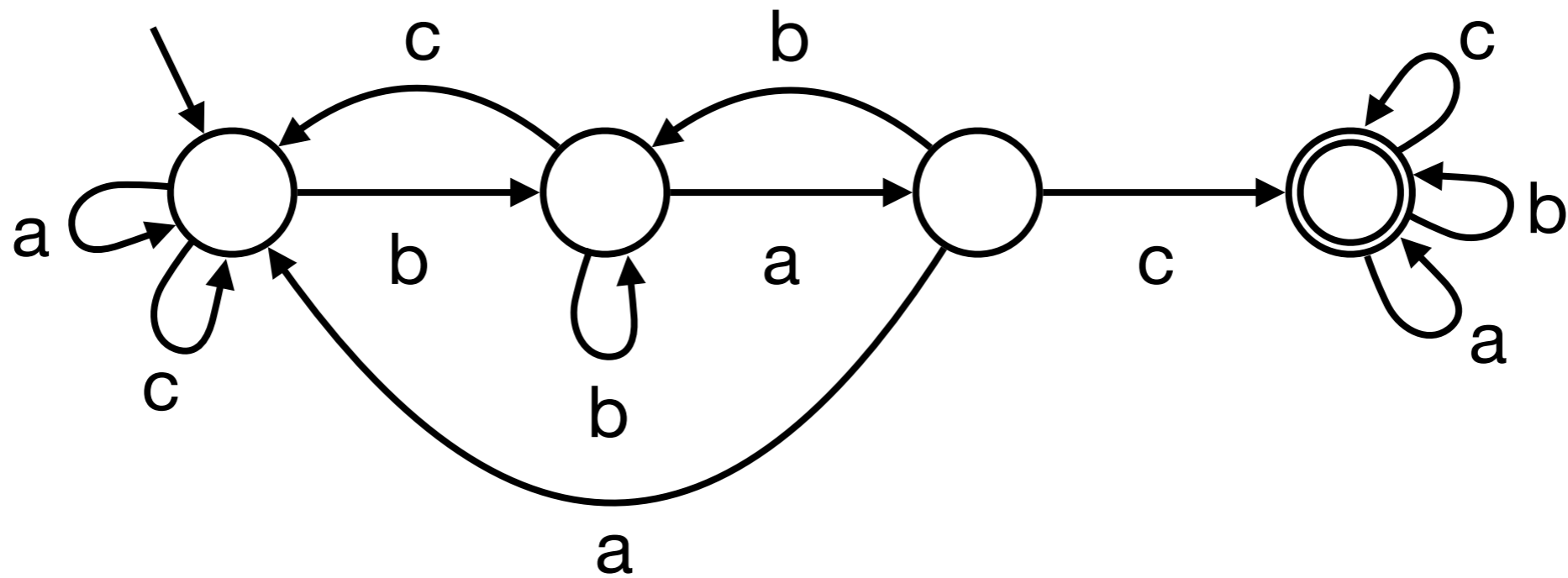
Exemples de recherche



abacaaa

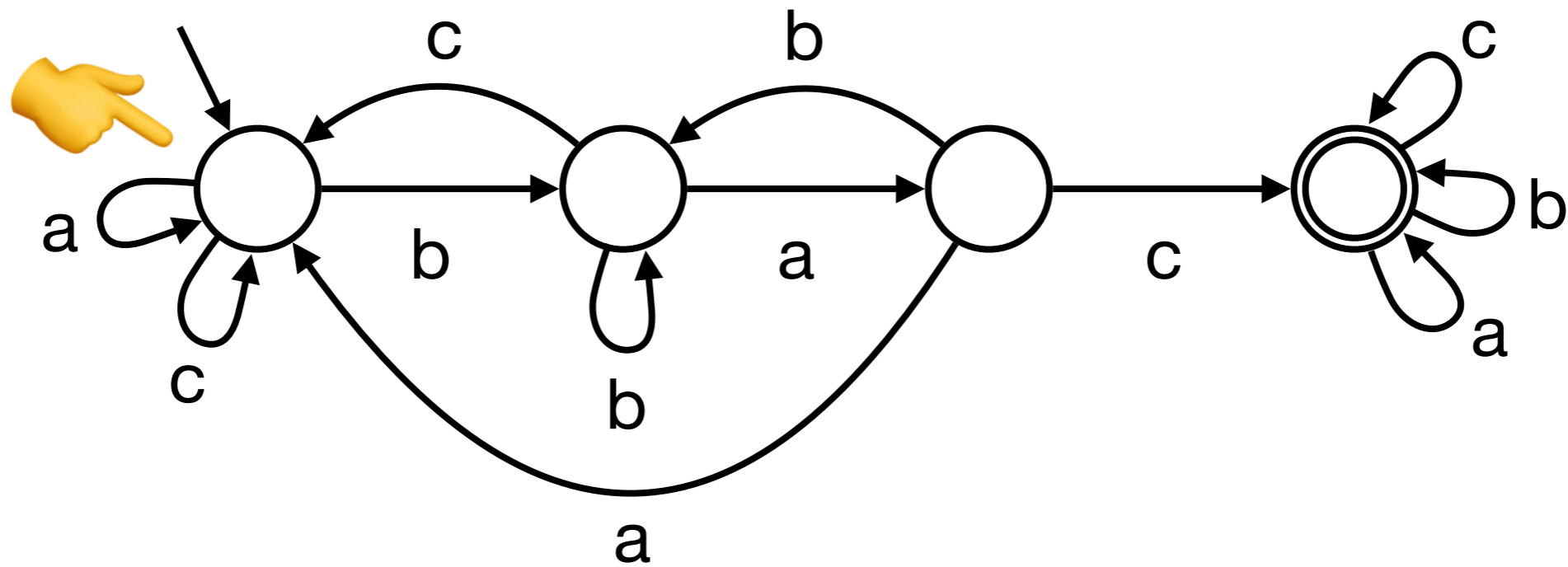


Exemples de recherche



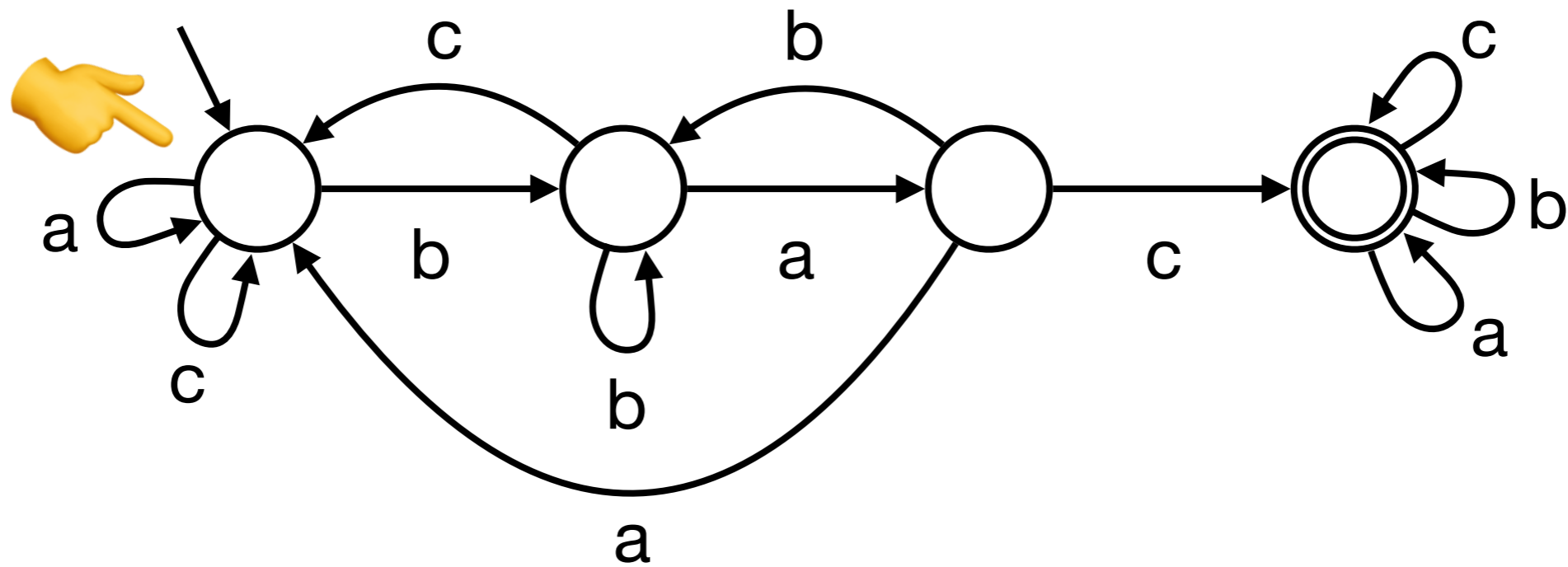
ababaac

Exemples de recherche



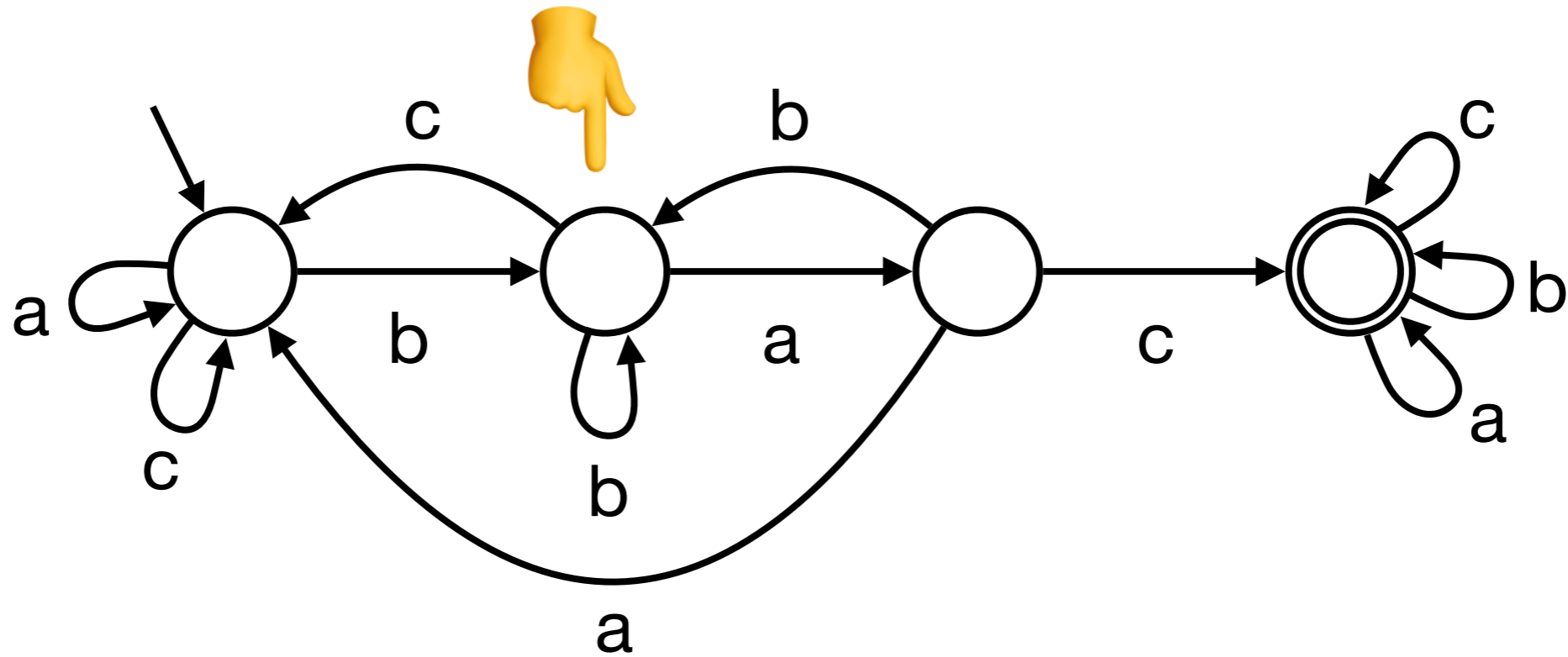
ababaac

Exemples de recherche



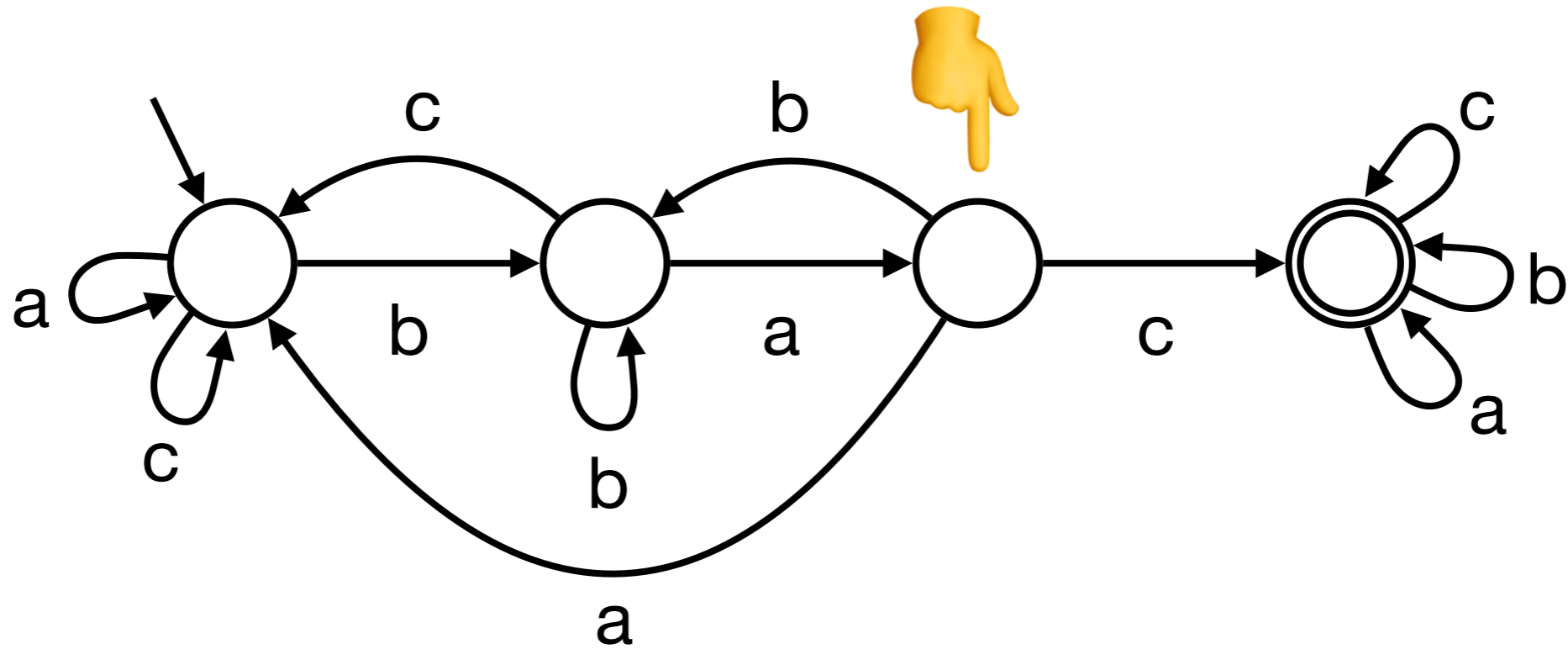
ababaac

Exemples de recherche



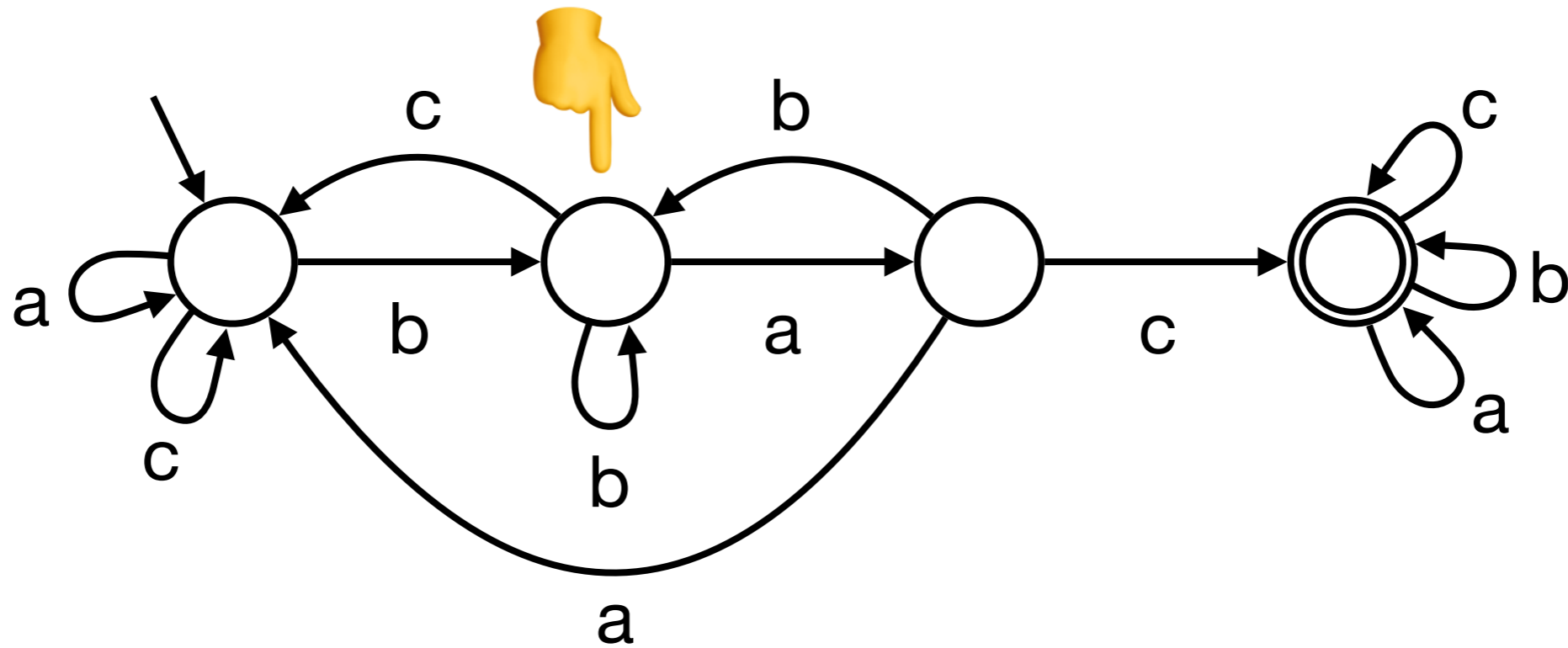
ababaac

Exemples de recherche



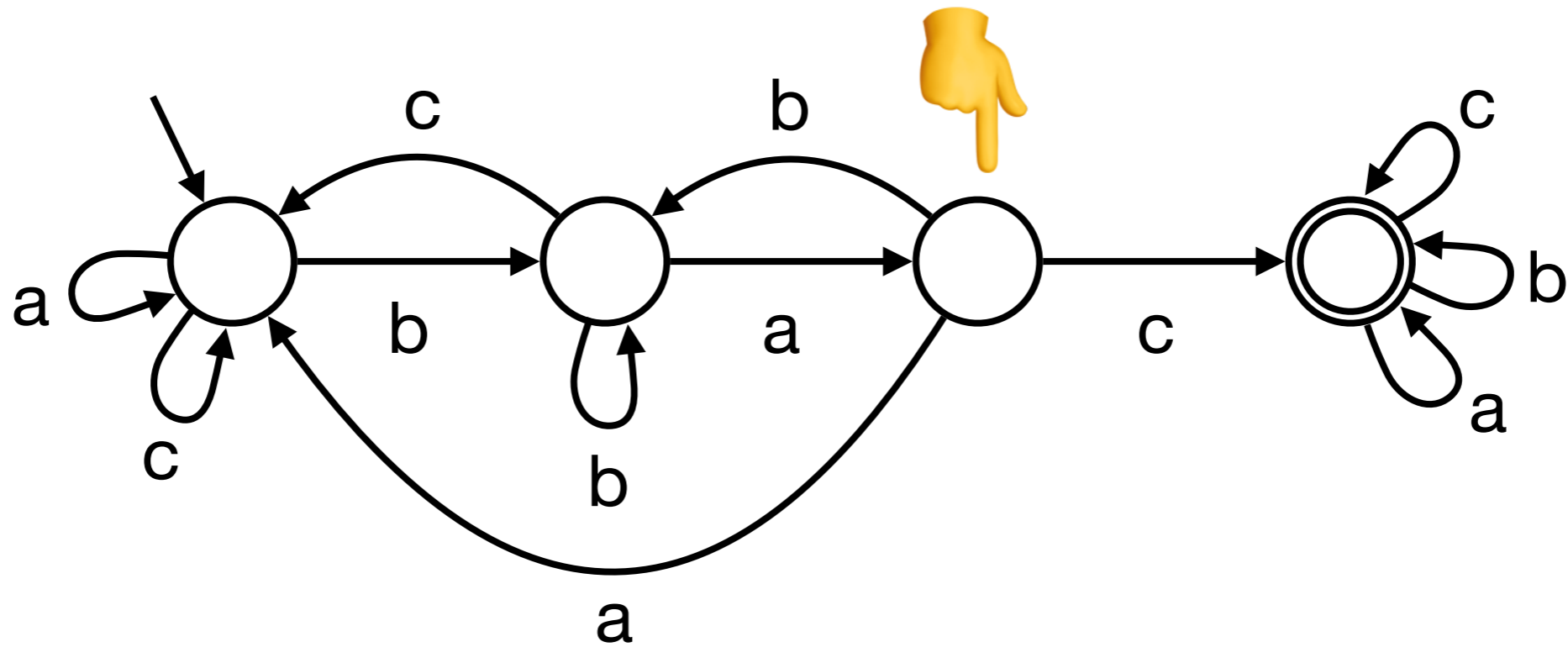
ababaac

Exemples de recherche



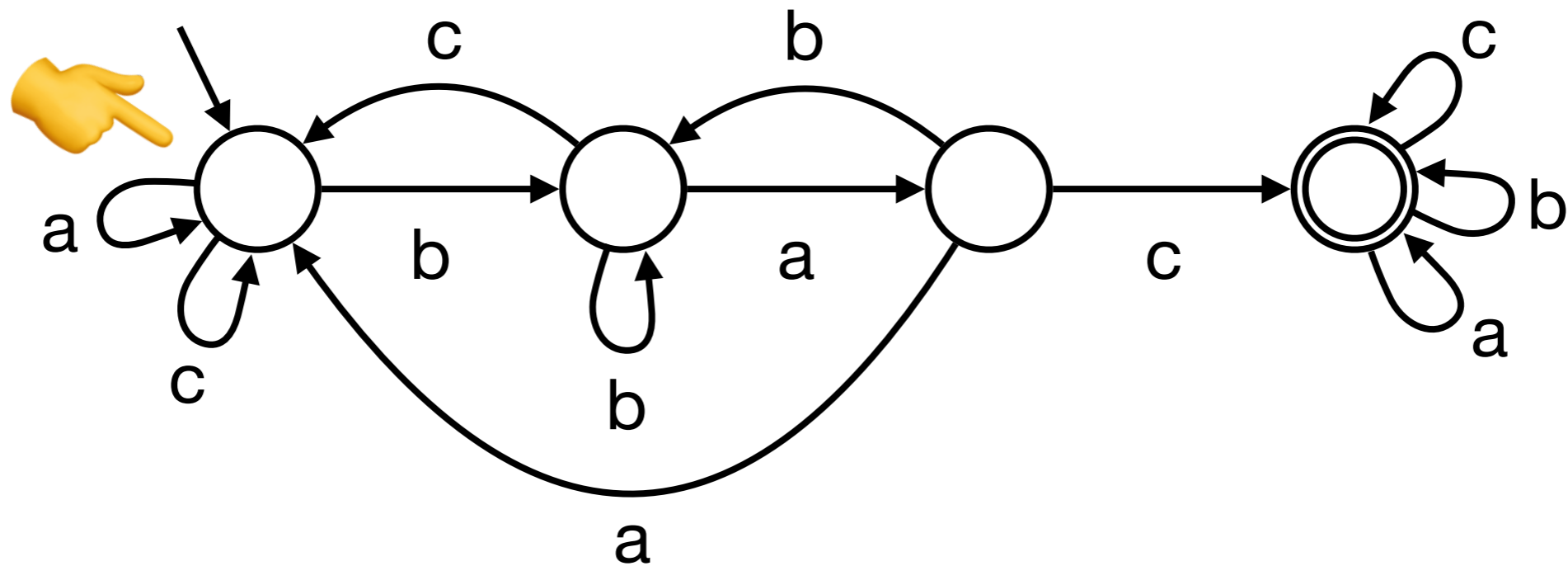
ababaac

Exemples de recherche



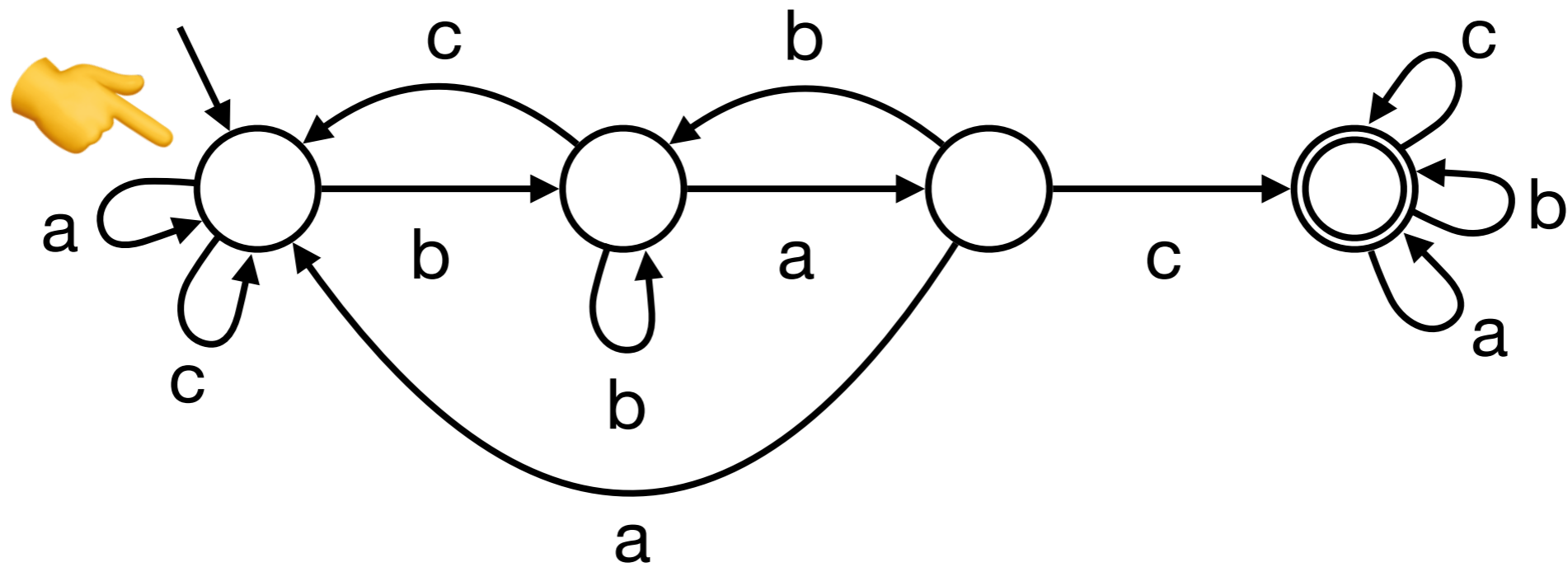
ababaac

Exemples de recherche



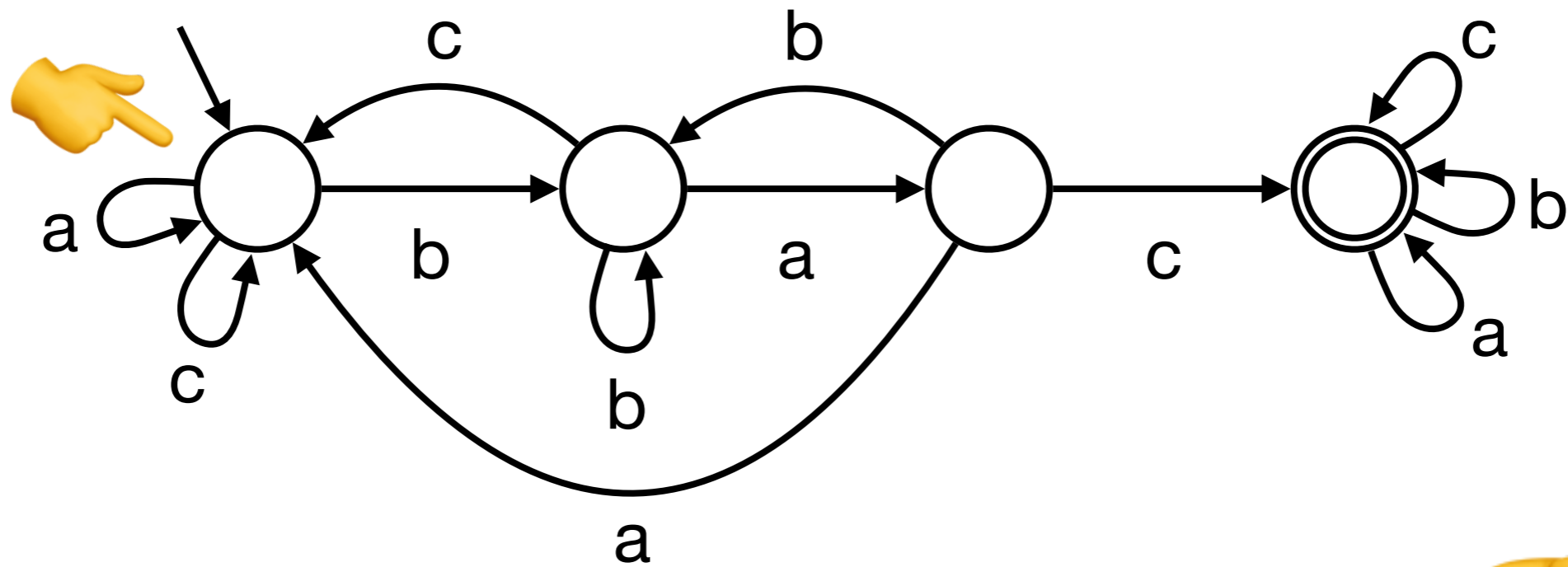
ababaac

Exemples de recherche



ababaac

Exemples de recherche



ababaac

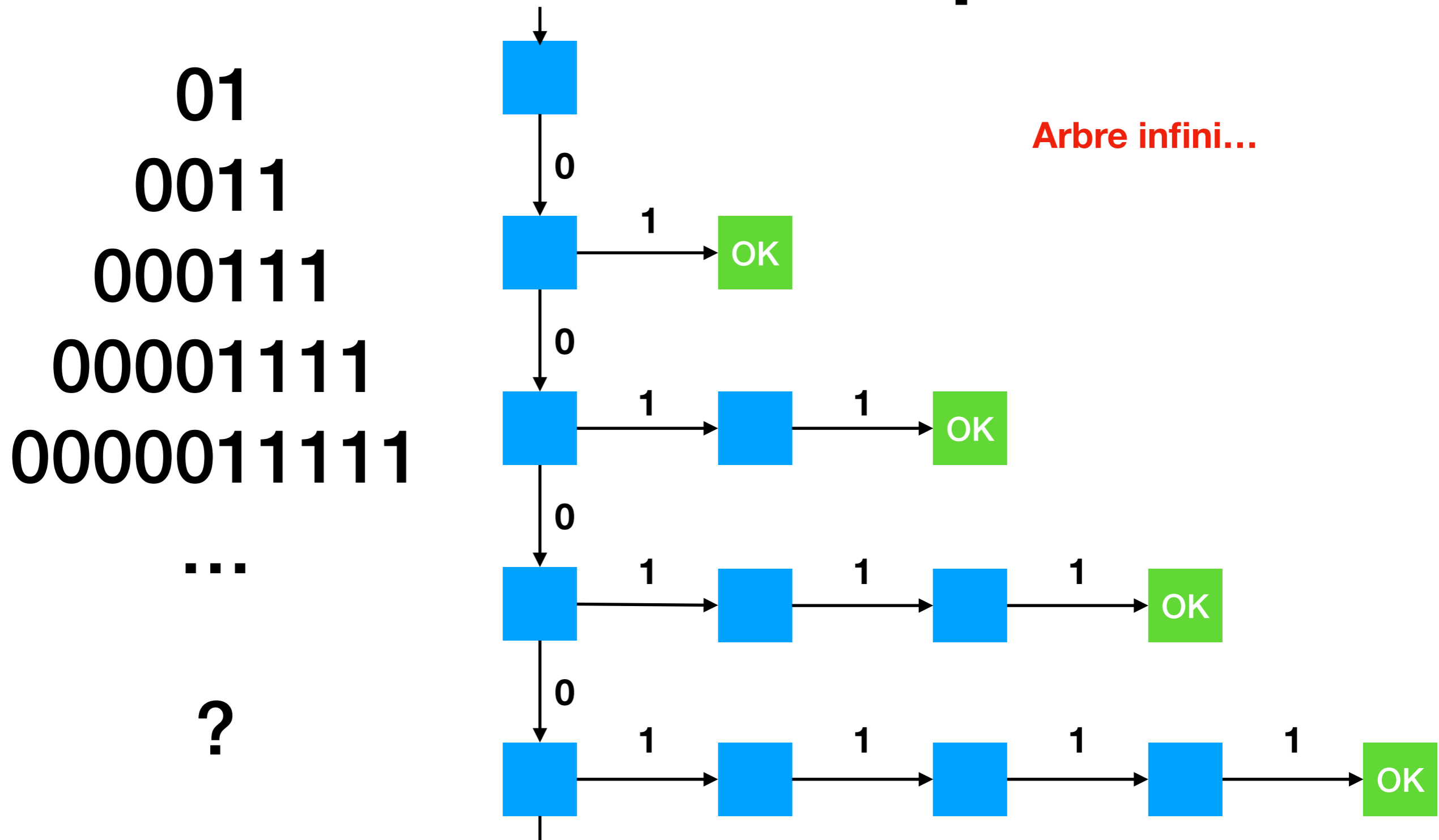
Exercice 4 du TD6

Limites des automates finis

Il n'existe pas un automate fini qui accepte exactement l'ensemble infini de mot suivant :

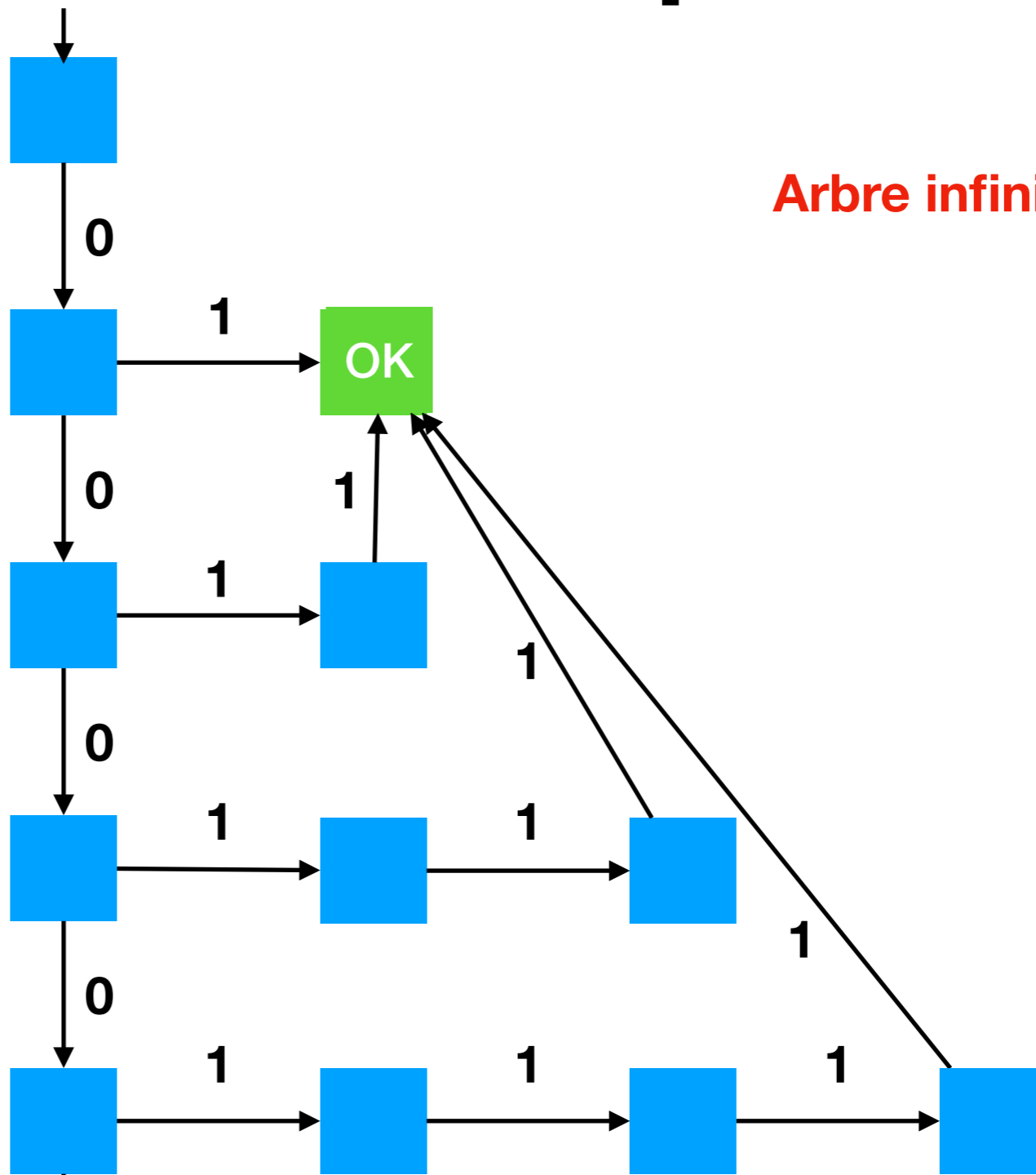
01
0011
000111
00001111
0000011111
000000111111
⋮

Comment peut-on accepter l'ensemble des séquences



Comment peut-on accepter l'ensemble des séquences

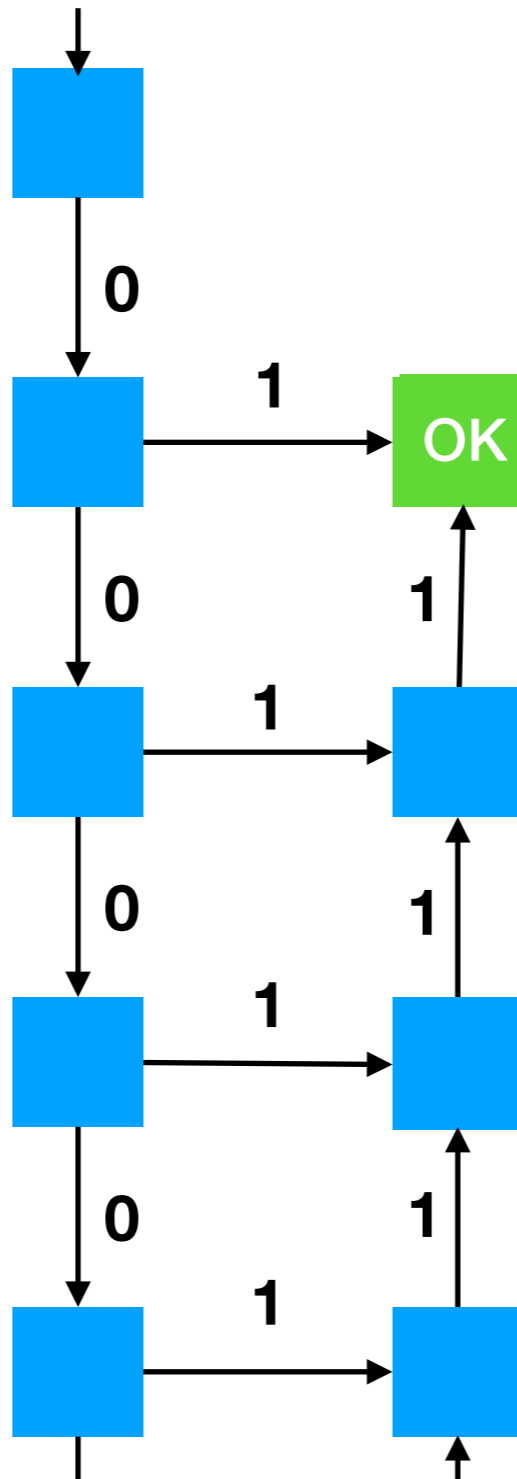
01
0011
000111
00001111
0000011111
...
?



Arbre infini...

Comment peut-on accepter l'ensemble des séquences

01
0011
000111
00001111
0000011111
...
?



Arbre infini...

... qu'on transforme en graphe infini...

Mais on ne peut pas le transformer en automate fini !

Comment peut-on accepter l'ensemble des séquences

01

0011

000111

00001111

0000011111

...

?

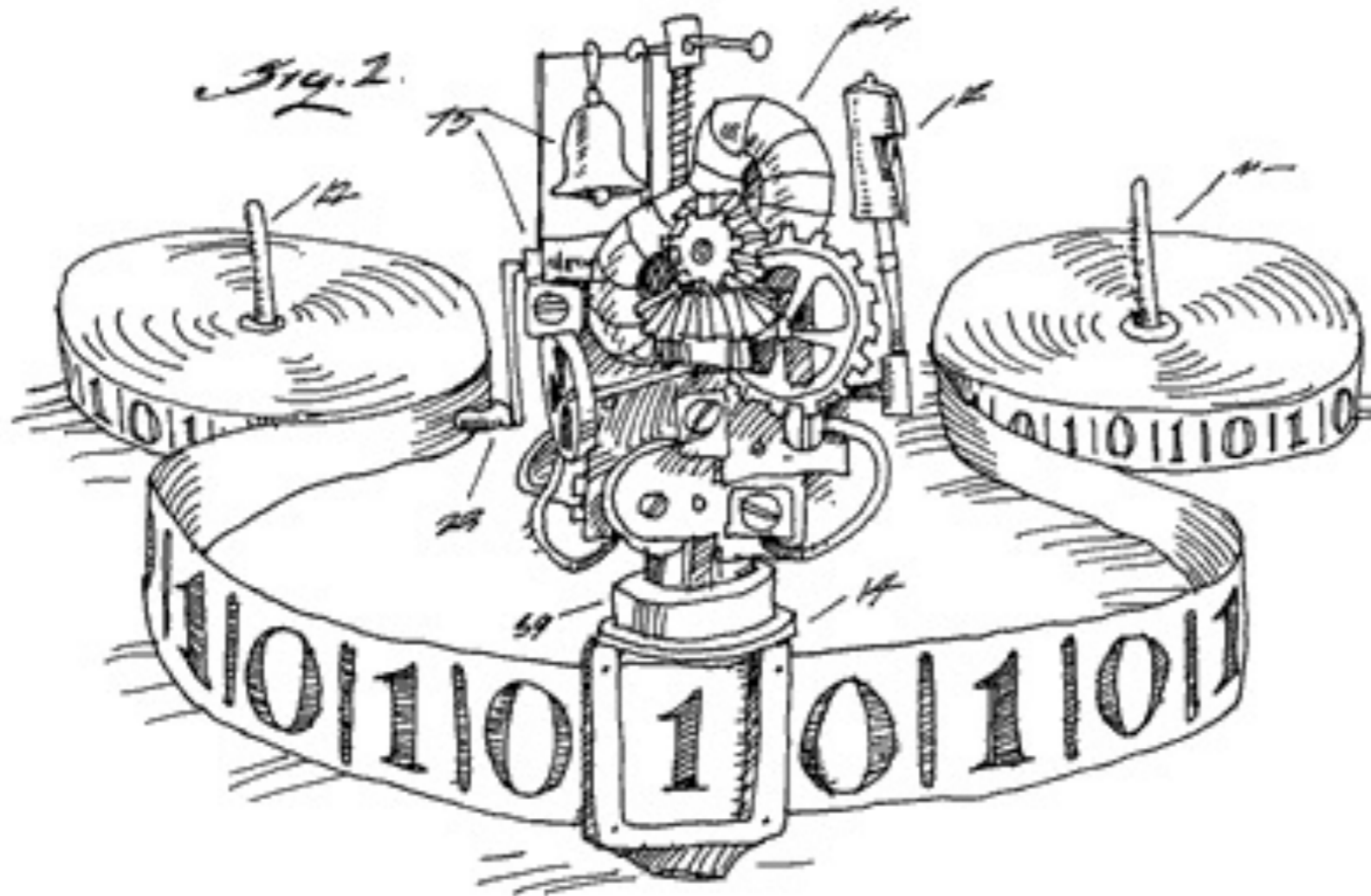
Et pourtant, il existe un algorithme pour décider si une séquence a cette forme !

1. Vérifier qu'il y a un bloc de 0 suivi d'un bloc de 1
2. Compter le nombre de 0 : n
3. Compter le nombre de 1 : p
4. Vérifier que $n = p$

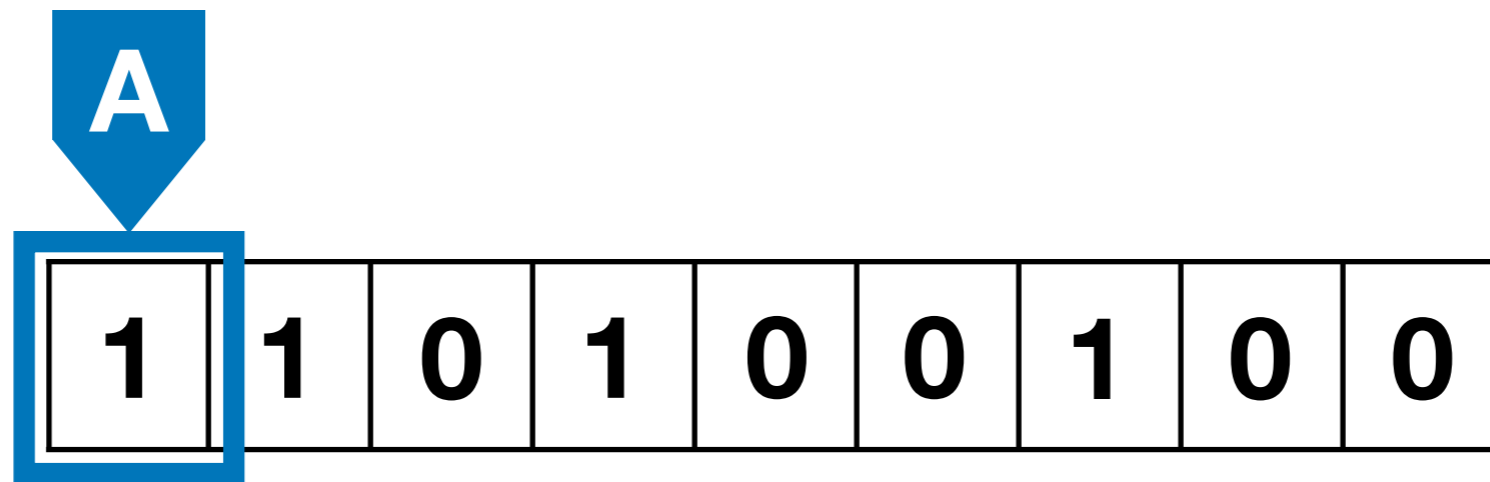
Les automates finis ne sont donc pas un modèle suffisant pour dire ce qui est « calculable »

Machines de Turing

Alan Turing (1912-1954)



Des automates...

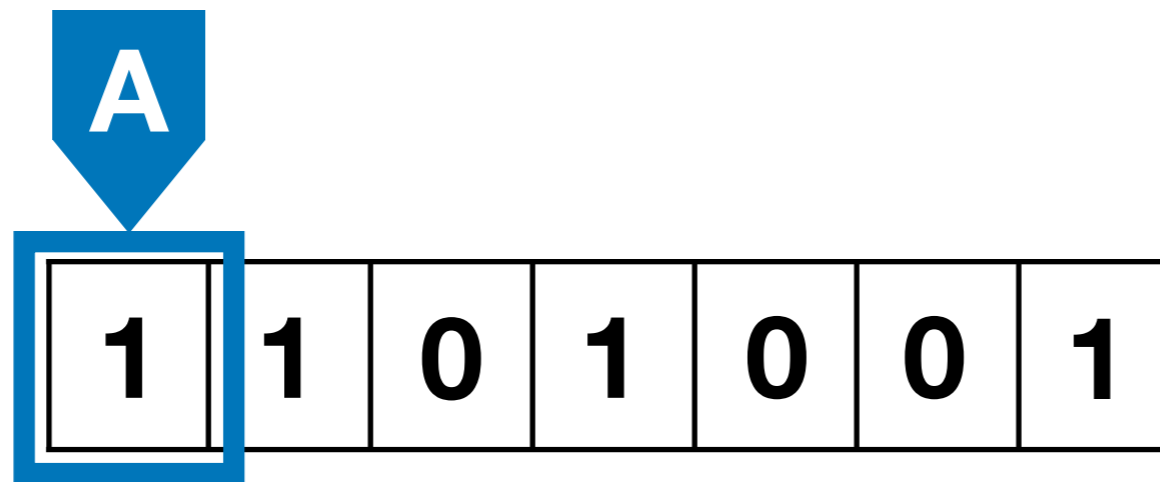


OK

```
Si état = A et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = A et on lit 0 alors
    état := B
    se déplacer à droite
Sinon Si état = B et on lit 1 alors
    état := A
    se déplacer à droite
Sinon Si état = B et on lit 0 alors
    état := B
    se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

Des automates...

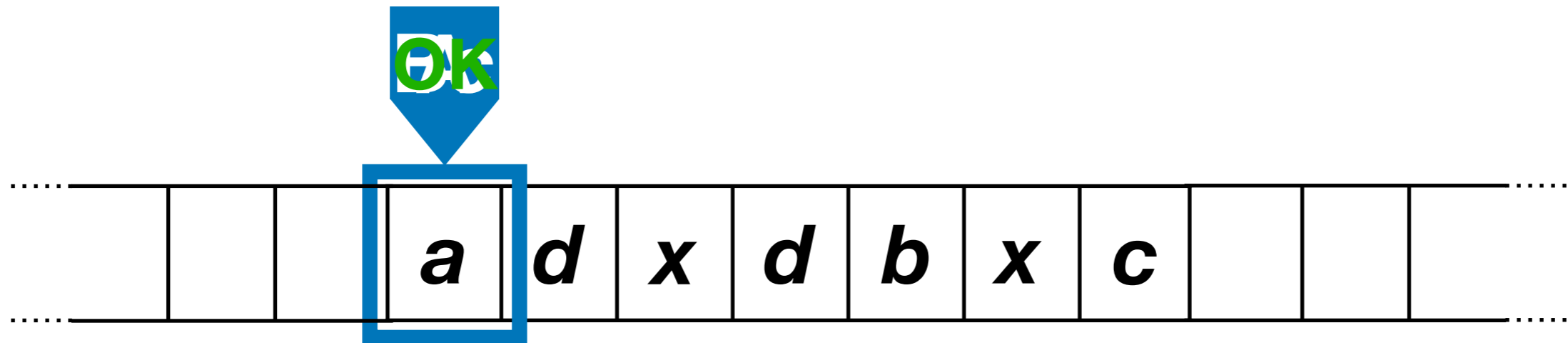


pas OK

```
Si état = A et on lit 1 alors
  état := A
  se déplacer à droite
Sinon Si état = A et on lit 0 alors
  état := B
  se déplacer à droite
Sinon Si état = B et on lit 1 alors
  état := A
  se déplacer à droite
Sinon Si état = B et on lit 0 alors
  état := B
  se déplacer à droite
Sinon s'arrêter !
FinSi
```

Etat	Symbole	Nouvel état
A	1	A
A	0	B
B	1	A
B	0	B

... qui peuvent changer de sens ...



Accepte si le dernier symbole de la séquence est différent de tous les autres symboles

Faisable avec un très gros automate...

Etat	Symbole	Sens	Nouvel état
A	a, b, c, \dots, z	→	A
A		←	B
	a	←	Ba
	b	←	Bb

Si état = A et on lit a alors
état := A
se déplacer à droite
Sinon Si état = A et on ne lit rien alors
état := B
se déplacer à gauche

...

BC

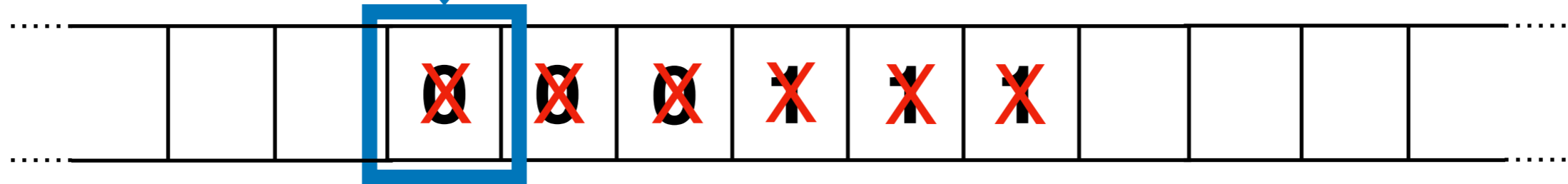


OK

... et écrire sur la bande

0100100

OK



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
Droite0	0	→	0	Droite1
Droite1	0	→	0	Droite1
Droite1	1	→	1	Droite1
Droite1	0	→	X	Gauche1
Droite1	X	→	X	Gauche0
Droite1	1	→	1	Gauche0
Gauche0	0	←	0	Gauche0
Gauche0	0	→	0	Droite0
Droite1	X	←	X	Gauche1
Droite0	X	→	X	OK

Si état = Droite0 et on lit 0 alors
état := Droite1
écrire ~~X~~ sur la bande
se déplacer à droite
...

Machine de Turing = calculateur humain avec papier et crayon



NACA (Comité consultatif national pour l'aéronautique), USA, 1950s

« Normalement on calcule en écrivant certains symboles sur le papier. [...] Je considère qu'on effectue le calcul sur un **papier unidimensionnel**, c'est-à-dire, sur un **ruban divisé en carrés**. »

– Alan M. Turing, *On computable numbers*

Papier 2D vs ruban 1D

A	B	C
D	E	F
G	H	I
J	K	L

A	B	C	;	D	E	F	;	G	H	I	;	J	K	L
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Papier 2D vs ruban 1D

A	B	C
D	E	F
G	H	I
J	K	L

M	N	O
P	Q	R
S	T	U
V	W	X

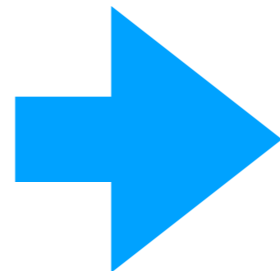
A	B	C	;	D	E	F	;	G	H	I	;	J	K	L	:	M	N	O	P	Q	R	S	T	U	V	W	X
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

« Je suppose aussi que le **nombre de symboles** qu'on peut écrire est **fini**. Si on permettait une infinité de symboles, il y aurait des symboles qui diffèrent dans une mesure arbitrairement faible [...] **On peut toujours utiliser une séquence de symboles au lieu d'un symbole simple.** »

– Alan M. Turing, *On computable numbers*

Symboles atomiques vs composites

1982



1	9	8	2
---	---	---	---

« La différence, de notre point de vue, entre les symboles simples et composites est qu'**on ne peut pas observer les symboles composites en un coup d'œil**, s'ils sont trop longs. Cela est conforme à l'expérience. On ne peut pas établir en un coup d'œil si 99999999999999999999 et 99999999999999999999 sont égaux. »

– Alan M. Turing, *On computable numbers*

« Champ visuel »

0	1	1	0	0	1	1	0		1	0	0	1	0	0
---	---	---	---	---	---	---	---	--	---	---	---	---	---	---

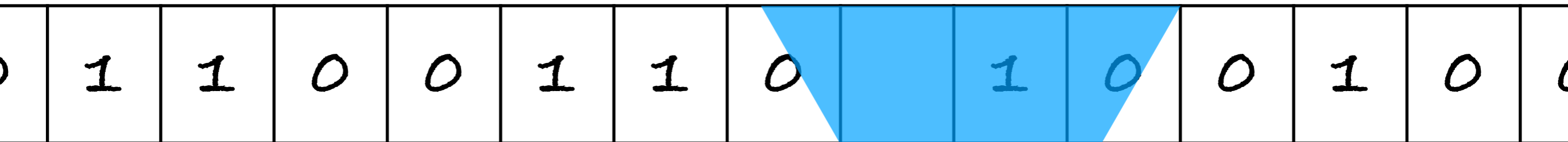


« Champ visuel »

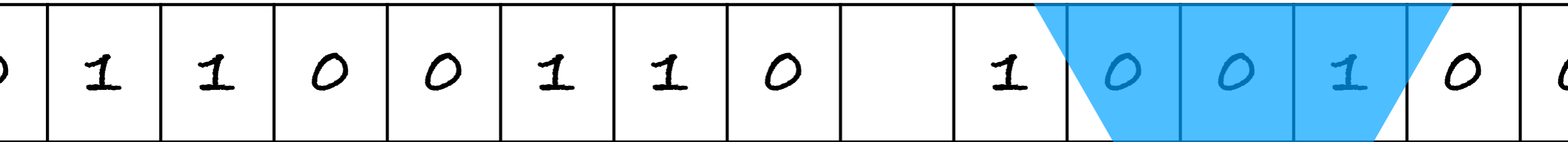
0	1	1	0	0	1	1	0	1	0	0	1	0	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---



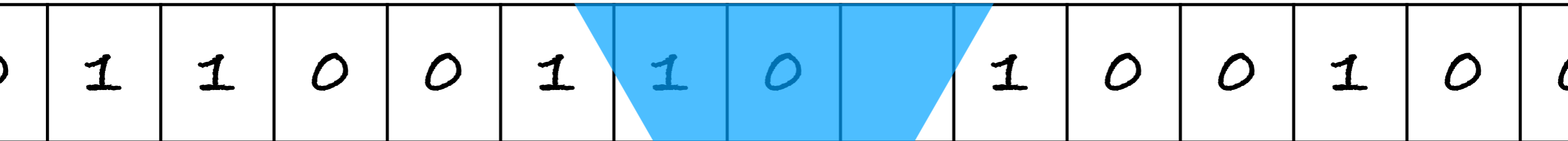
« Champ visuel »



« Champ visuel »



« Champ visuel »



« Champ visuel »

0	1	1	0	0	1	1	0	1	0	0	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---

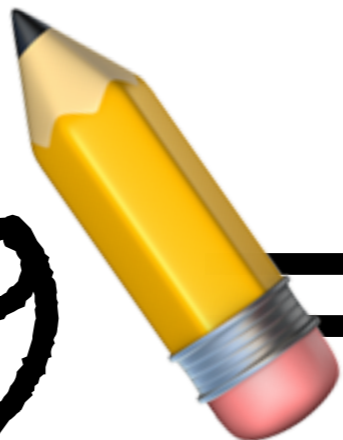


« Le **comportement** du calculateur à chaque moment est **déterminé par le symbole qu'il observe et son "état d'esprit"** à ce moment. »

– Alan M. Turing, *On computable numbers*

« États d'esprit »

12932 +
19 =



J'ai lu le
chiffre 2



« États d'esprit »

12932 +

19 =

J'ai lu le
chiffre 2 et le
chiffre 9



« États d'esprit »

$$12932 + 19 =$$

Il faut que j'écrive 1 et que je garde 1 comme retenue



« États d'esprit »

$$\begin{array}{r} 12932 \\ + \\ 19 \\ \hline \end{array}$$

Il faut que je
me déplace à
gauche ; la
retenue est 1

1



« États d'esprit »

12932

+

19

=

J'ai lu le
chiffre 3 ; avec la
retenue de 1 ça
fait 4

1



« États d'esprit »

12932 +

19 =

1

J'ai lu 4 et le
chiffre 1



« États d'esprit »

$$12932 + 19 =$$

Il faut que
j'écrive 5 ; pas
de retenue

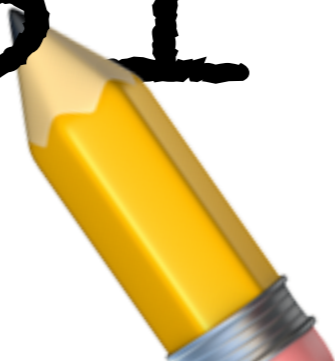


« États d'esprit »

$$12932 + 19 =$$

Il faut que je
me déplace à
gauche

51



« On suppose également que le **nombre d'états d'esprit** qu'on doit prendre en compte est **fini**. Les raisons pour cela sont de la même nature que celles qui restreignent le nombre de symboles. »

– Alan M. Turing, *On computable numbers*

États d'esprit trop proches

J'ai lu la
séquence
9999999999



États d'esprit trop proches

J'ai lu la
séquence
9999999999

J'ai lu la
séquence
9999999999



« On peut éviter l'utilisation d'états d'esprit plus compliqués **en écrivant plus de symboles** sur le ruban. »

– Alan M. Turing, *On computable numbers*

Prendre note sur le ruban

Le résultat
partiel est
9999999999



Prendre note sur le ruban

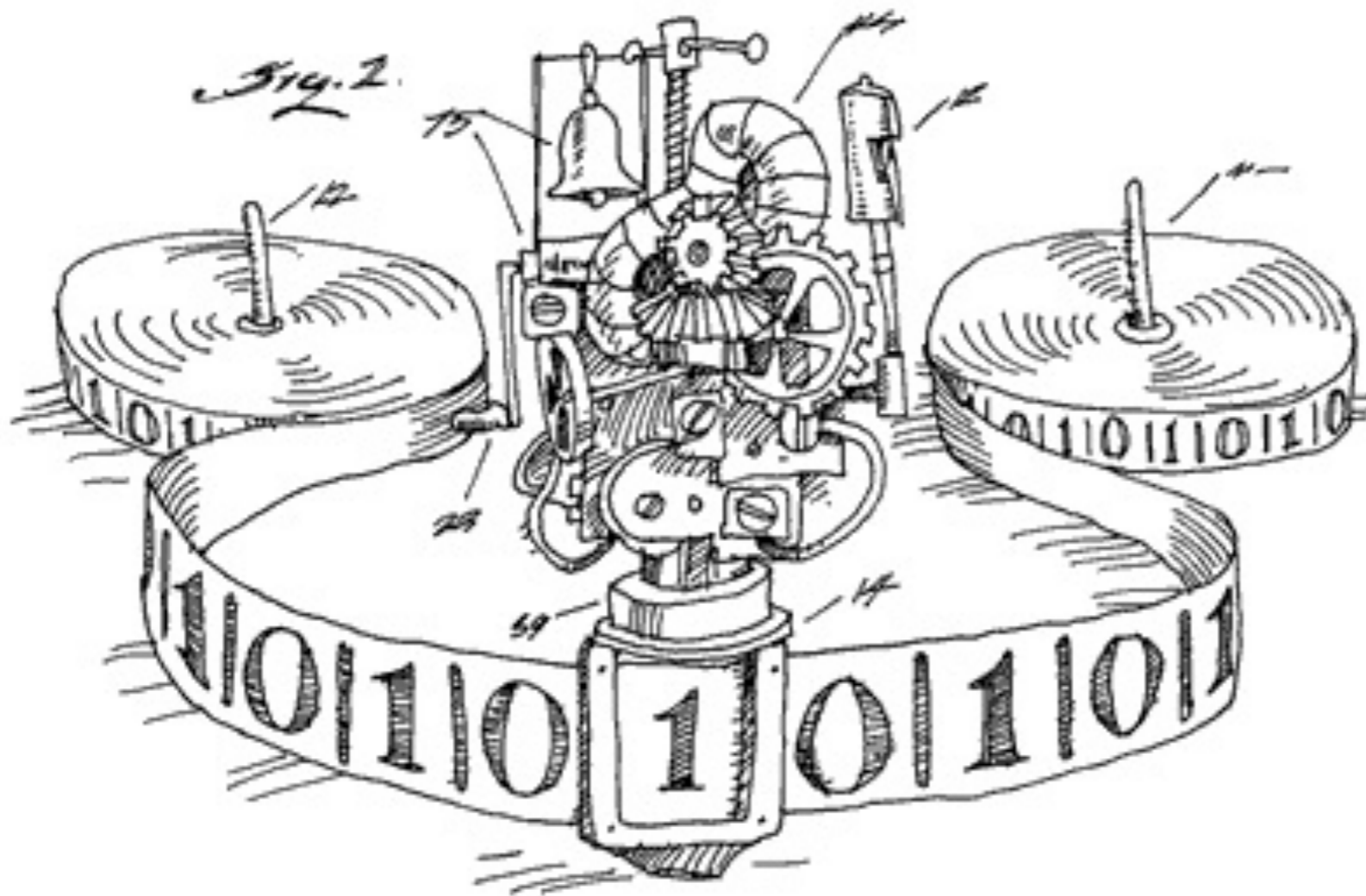
~~Le résultat
partiel est
999 999 999~~

Le résultat
partiel est écrit
sur le ruban



Machines de Turing

Alan Turing (1912-1954)



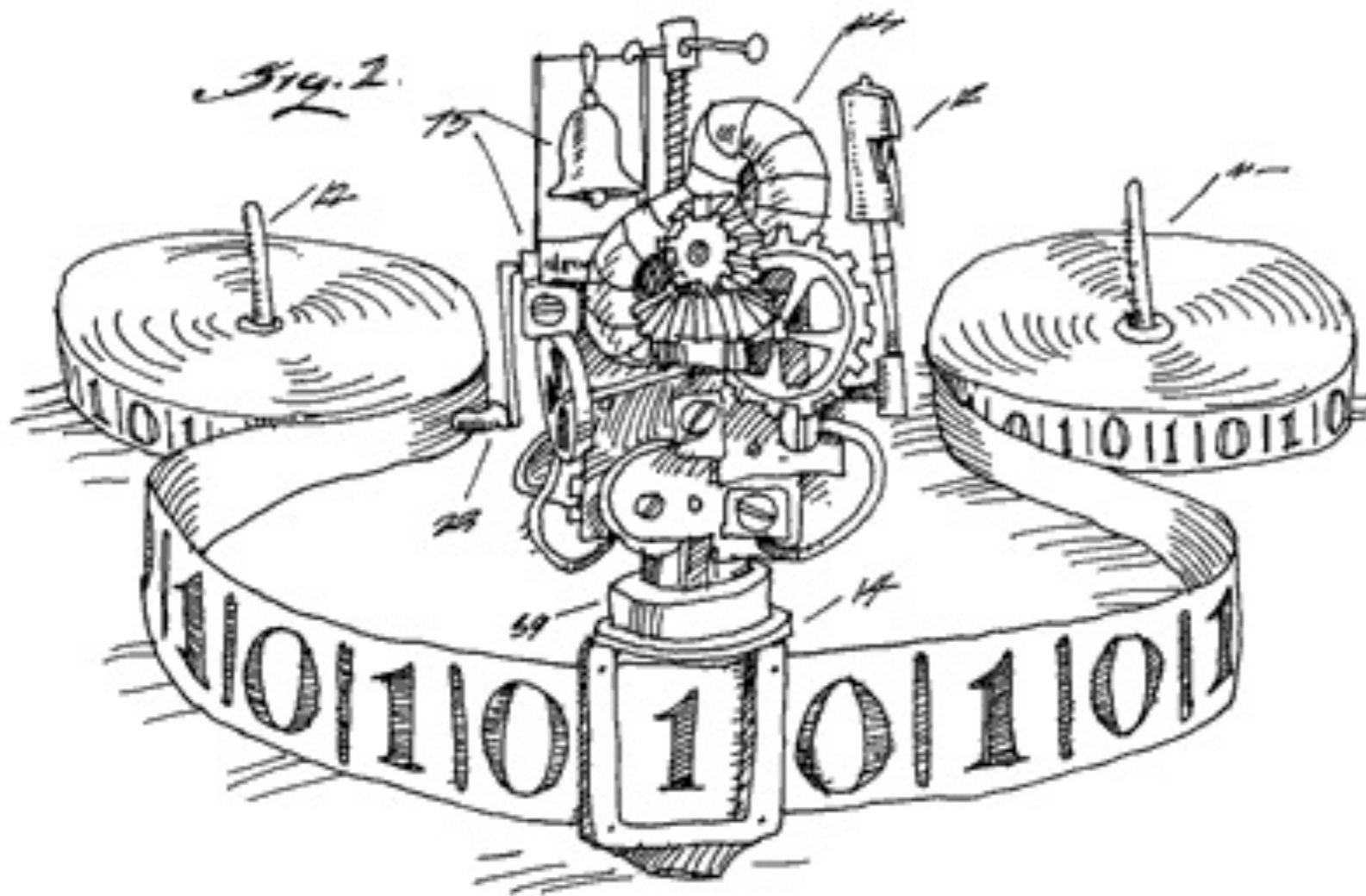
Des automates...

qui peuvent changer de sens...

et écrire sur la bande.

Machines de Turing

Alan Turing (1912-1954)



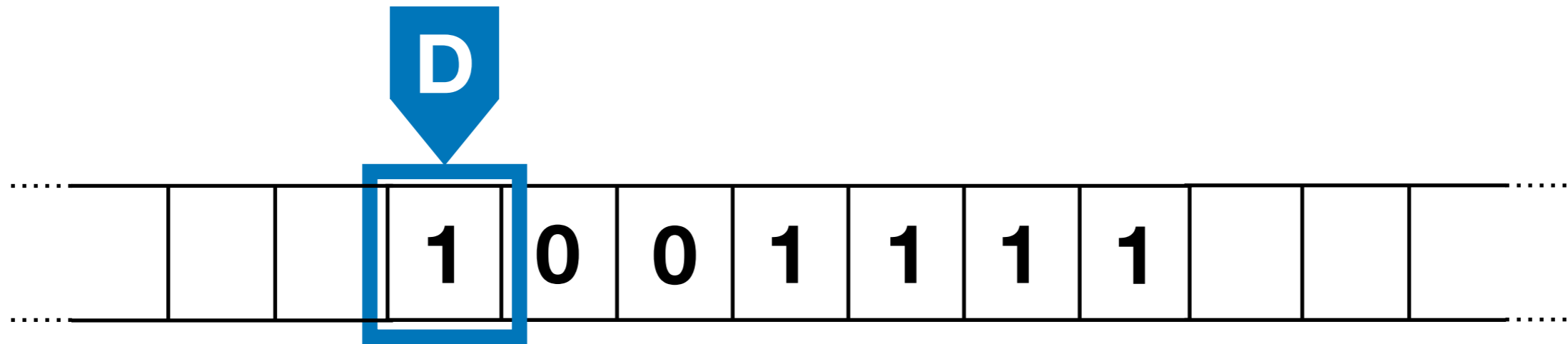
Des automates...

qui peuvent changer de sens...

et écrire sur la bande.

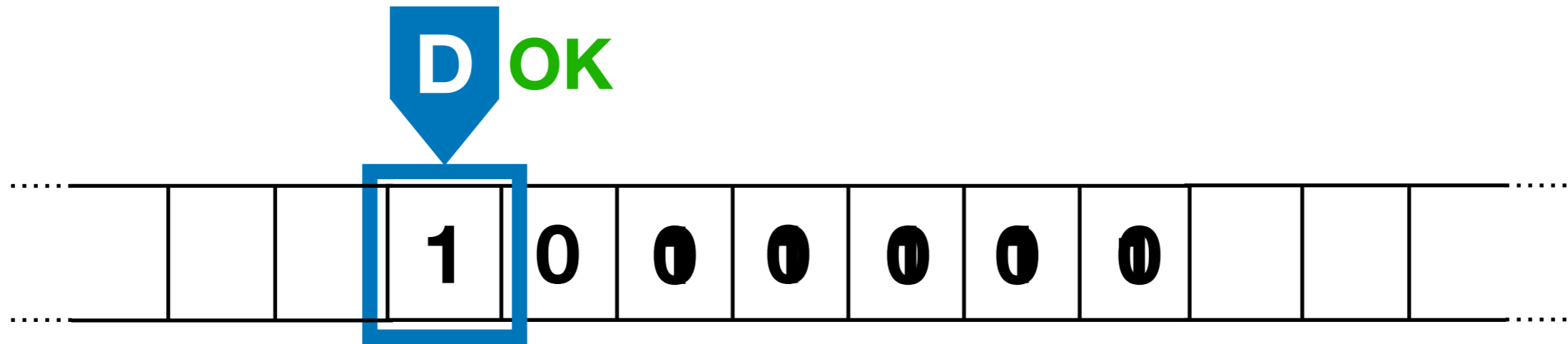
Exercice 1 du TD7

Que fait cette machine ?



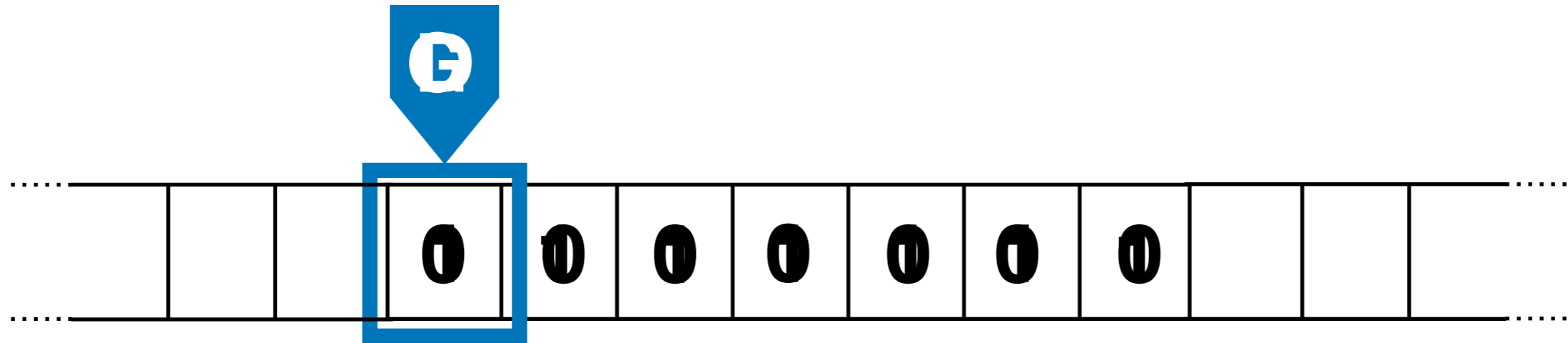
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



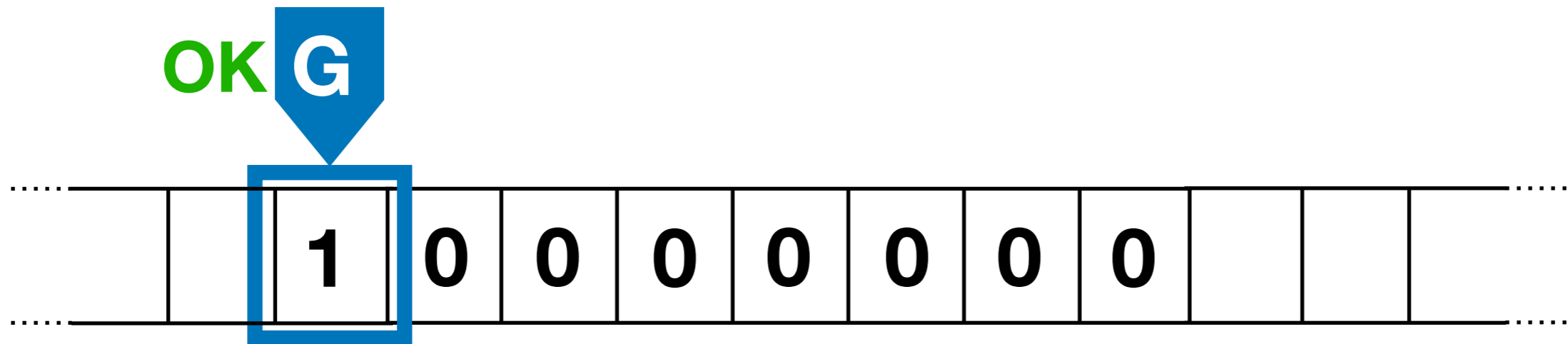
Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

Incrémentation



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK

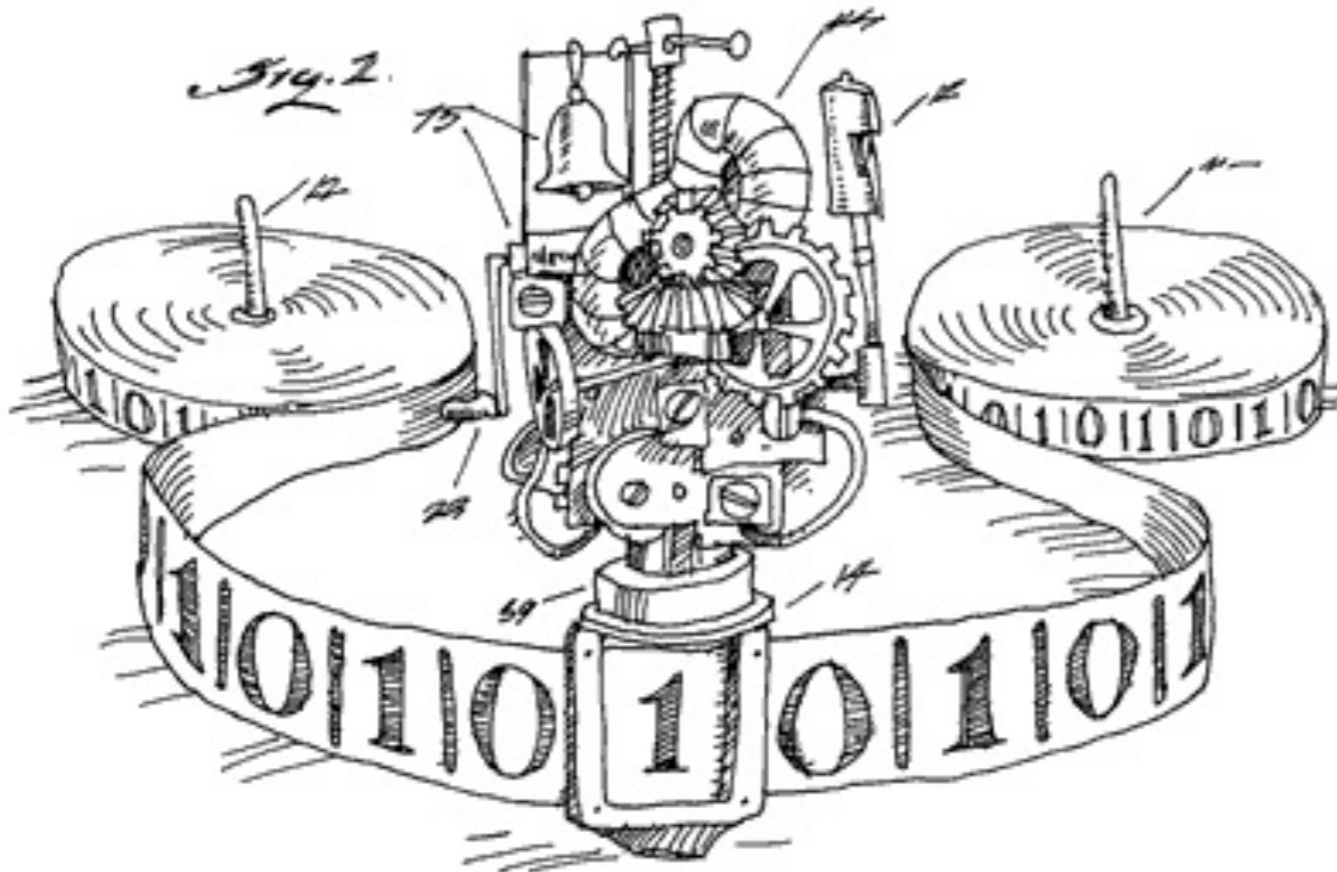
Incrémentation



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
D	0	→	0	D
D	1	→	1	D
D		←		G
G	1	←	0	G
G	0	←	1	OK
G		←	1	OK

Exercice 2 du TD7

**Machine de Turing =
Python =
ordinateur moderne**



Pseudo-code → Machine de Turing

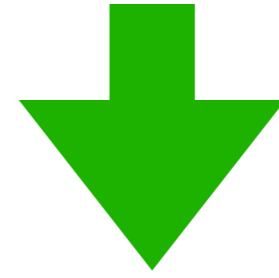
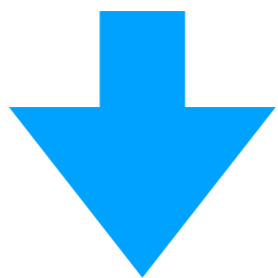
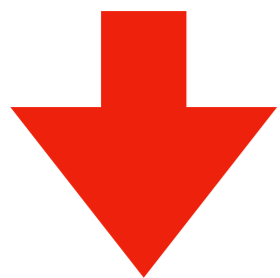
- Un peu d'arithmétique → l'incrémement par exemple !
- Séquence d'instructions → séquence d'états de la MT
- Conditions (**si... alors... sinon**) → table de transition
- Itération (boucles **tant que... faire, pour... faire**)
→ répétition d'états de la MT

$i = 5$

$j = 7$

$T =$

4	2	3	1	0	1	6	0
---	---	---	---	---	---	---	---



i	1	0	1	j	1	1	1	T	1	0	0	,	1	0	,	1	1	,
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Machine de Turing → Pseudo-code

Etat	Symbole	Sens	Nouveau symbole	Nouvel état
A	0	→	1	B
A	1	←	A	A
A		←	OK	B

Tant que état ≠ OK **faire**

Si état = A **et** on lit 0 **alors**

état := B

écrire 1 sur la bande

se déplacer à droite

Sinon Si état = A **et** on lit 1 **alors**

état := A

écrire 1 sur la bande

se déplacer à gauche

Sinon Si ...

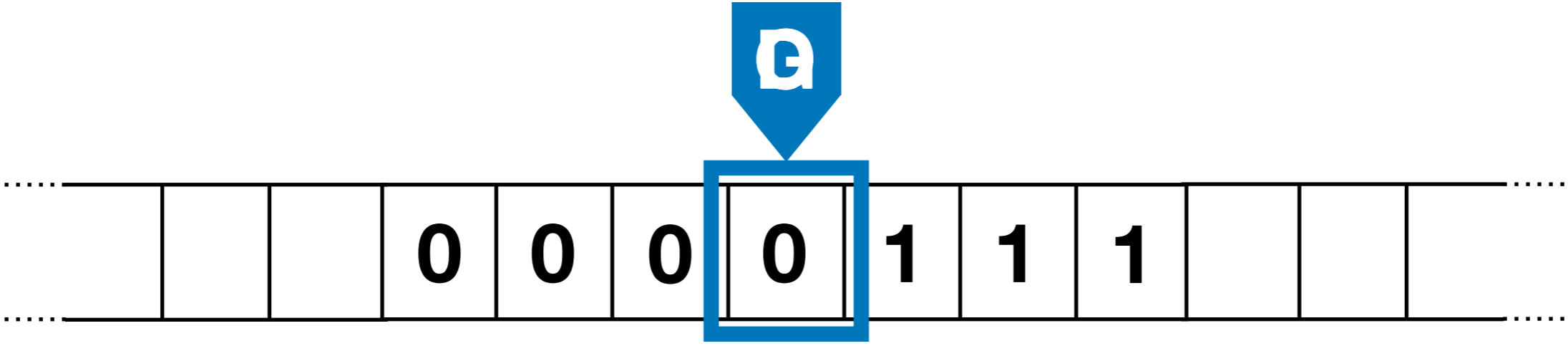
Certains codes ne terminent pas...

```
def p(n):  
    s = 0  
    i = 0  
    while i <= n:  
        s = s + i  
    return s
```

Que renvoie p(1) ?

Que renvoie p(42) ?

Certaines machines ne terminent pas...



Etat	Symbole	Sens	Nouveau symbole	Nouvel état
G		→	0	D
G	0	←	0	G
G	1	←	1	G
D		←	1	G
D	0	→	0	D
D	1	→	1	D