Turing Machines

FOCS, Fall 2020

Until now

- Decision functions
- Formal languages
- Finite state machines

A decision function $f : \Sigma^* \rightarrow \{true, false\}$ is FA-computable if there exists a finite state machines M such that f(u) = true exactly when M accepts string u

- f is FA-computable if the associated set A_f is regular

Not every language is regular — including some that correspond to easily computable functions (for us)

These languages are not regular

$\{a^n b^n \mid n \geq 0\} = \{ \varepsilon, ab, aabb, aaabbb, aaaabbbb, \ldots. \}$

$\{a^n b^n c^n \mid n \geq 0\}$

$\{a^n b^m \mid m \geq n \geq 0\}$

$\{u \in \{a,b\}^* \mid \#_a(u) = \#_b(u) \}$ $\#_x(u)$ = number of x in u

$\{u \in \{a,b\}^* \mid u = rev(u)\}$ $rev(u)$ = u reversed

Intuitively, finite state machines cannot remember an arbitrary amount of information (natural number = arbitrary amount of information - think digits)

Why is $\{a^n b^n \mid n \geq 0\}$ not regular?

We argue by contradiction. Assume $\{a^n b^n \mid n \geq 0 \}$ is regular. We derive a contradiction from that fact.

If $\{a^n b^n \mid n >= 0 \}$ is regular, there is a deterministic finite state machine that accepts it. Call it M. Machine M has some number K of states.

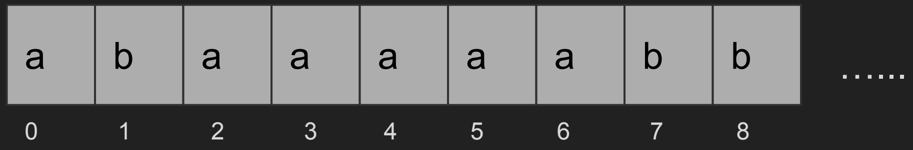
Consider the string $a^{K+1}b^{K+1}$. This string is accepted by M. When following a^{K+1} you go through K+1 states - so two of those states must be the same state S. So you have a loop of length L going from S to S. That loop forces M to also accept $a^{K+1+L}b^{K+1}$ by going around the loop one time more. And $a^{K+1+L}b^{K+1}$ is not in the language. This contradicts M accepting the language $\{a^n b^n \mid n \geq 0 \}$.

Our assumption was wrong: $\{a^n b^n \mid n \geq 0 \}$ is not regular.

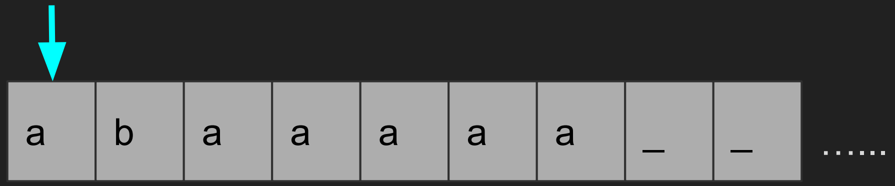
Adding memory to finite state machines

Let's give finite state machines some storage space (memory).

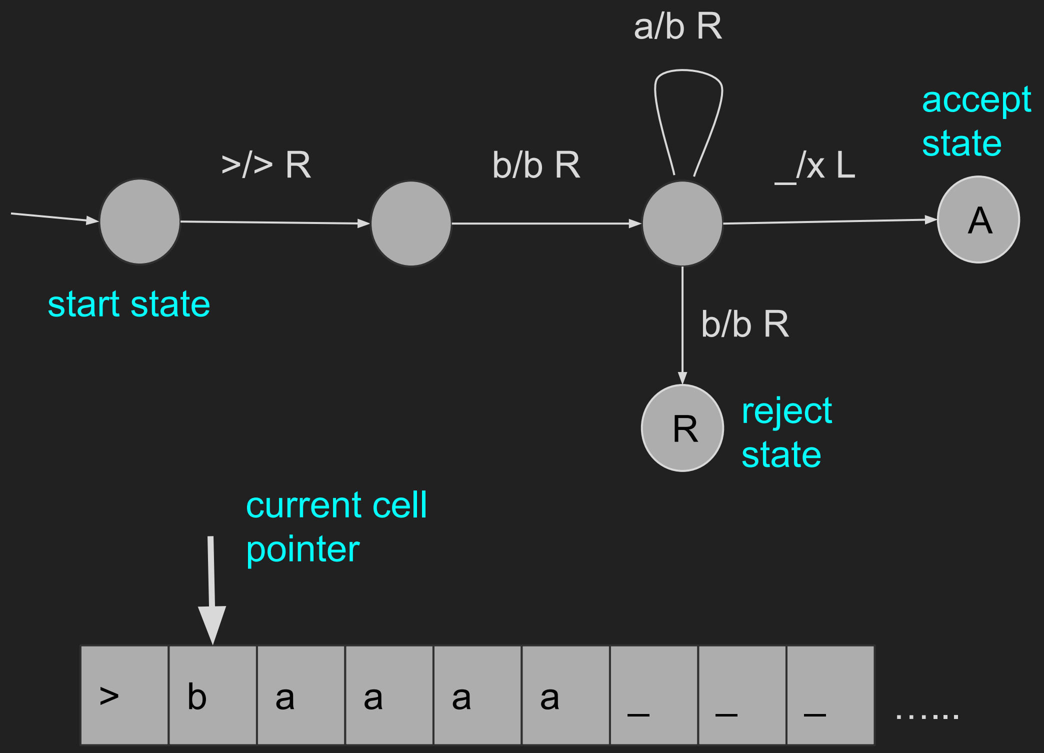
We're used to memory being a large array that you can index into (address):



That requires numbers, and numbers are complicated. Instead, we use a pointer that we can move left or right to go to a cell that we want to read/modify:



Turing Machines



tape initialized with > followed by input string

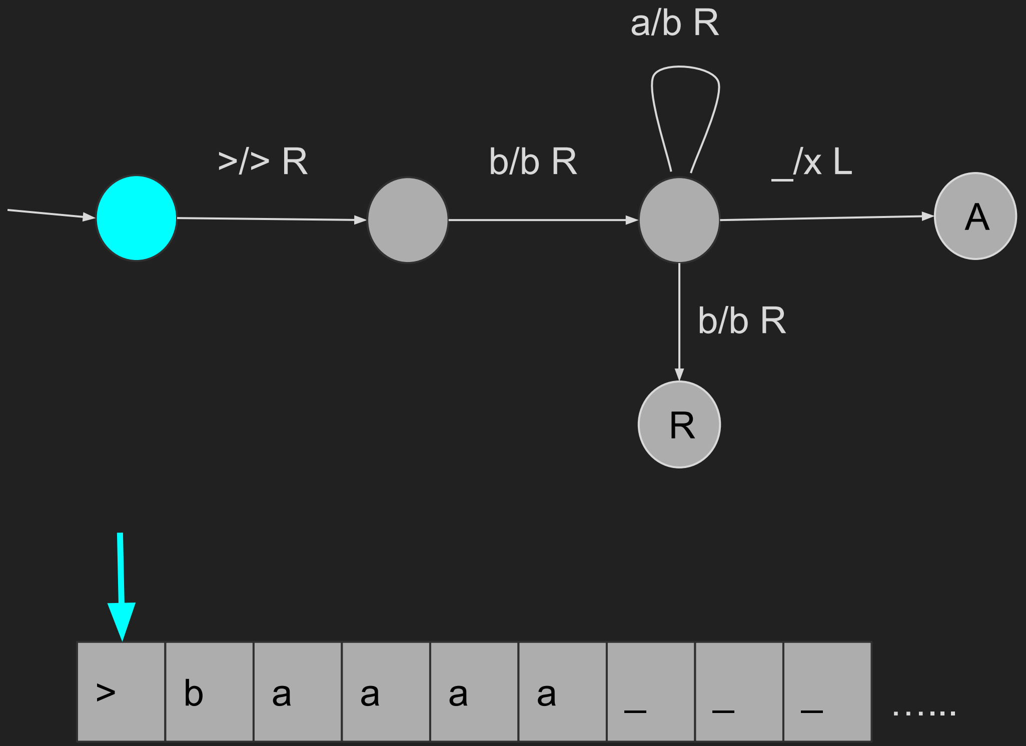
Finite set of states with labeled transitions between them

Labels of the form x/y D
"when on x, rewrite into y and
 move in direction D = L or R"

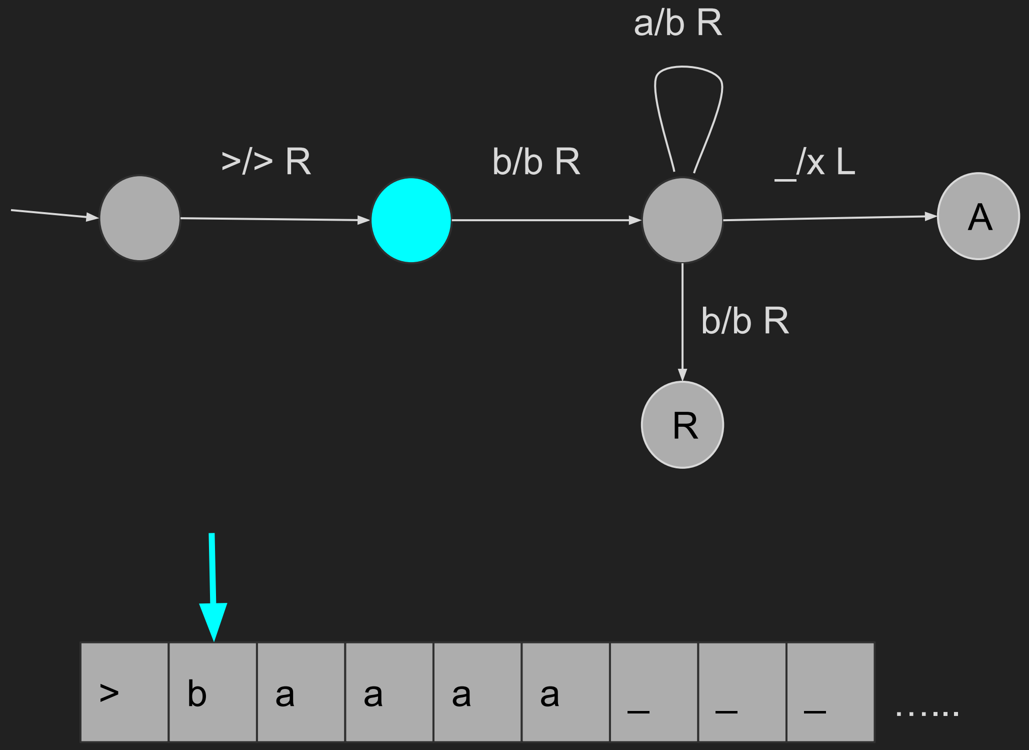
Deterministic - one transition with a given symbol out of every state

convention: if no transition with a given symbol, transition to reject state

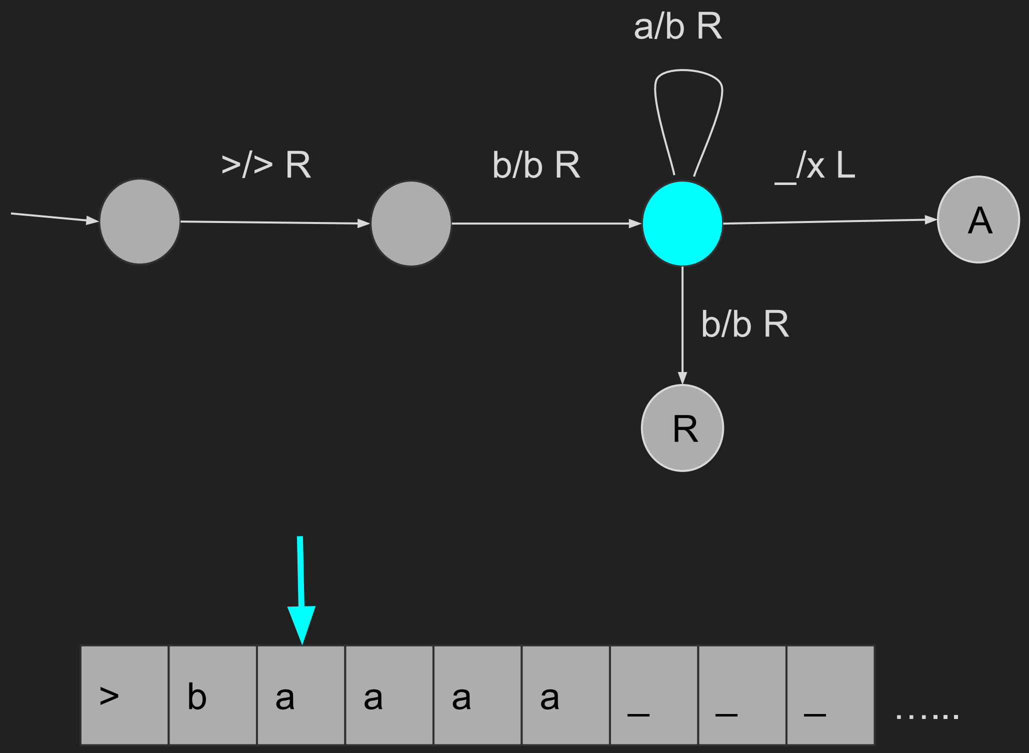
Running the Turing machine



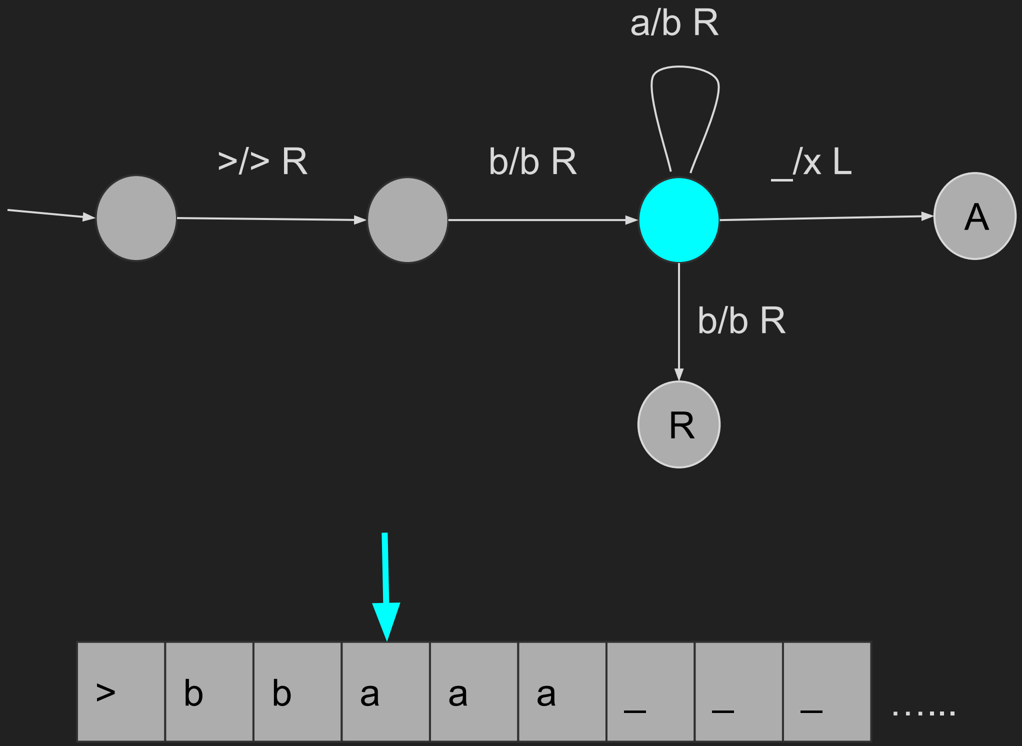
Running the Turing machine



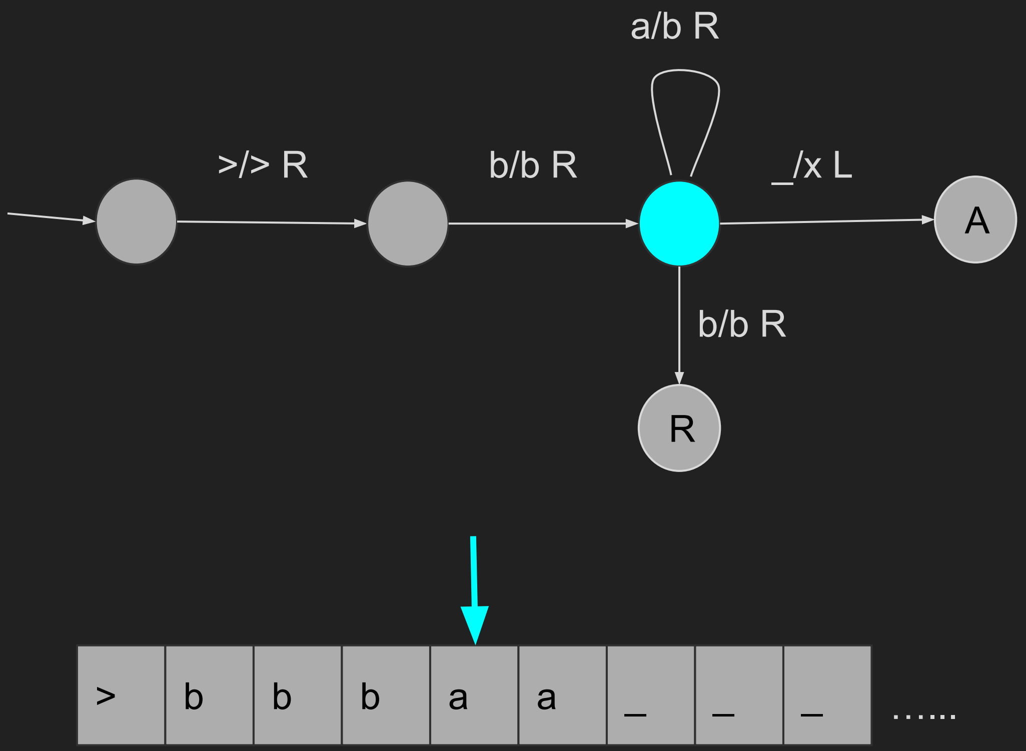
Running the Turing machine



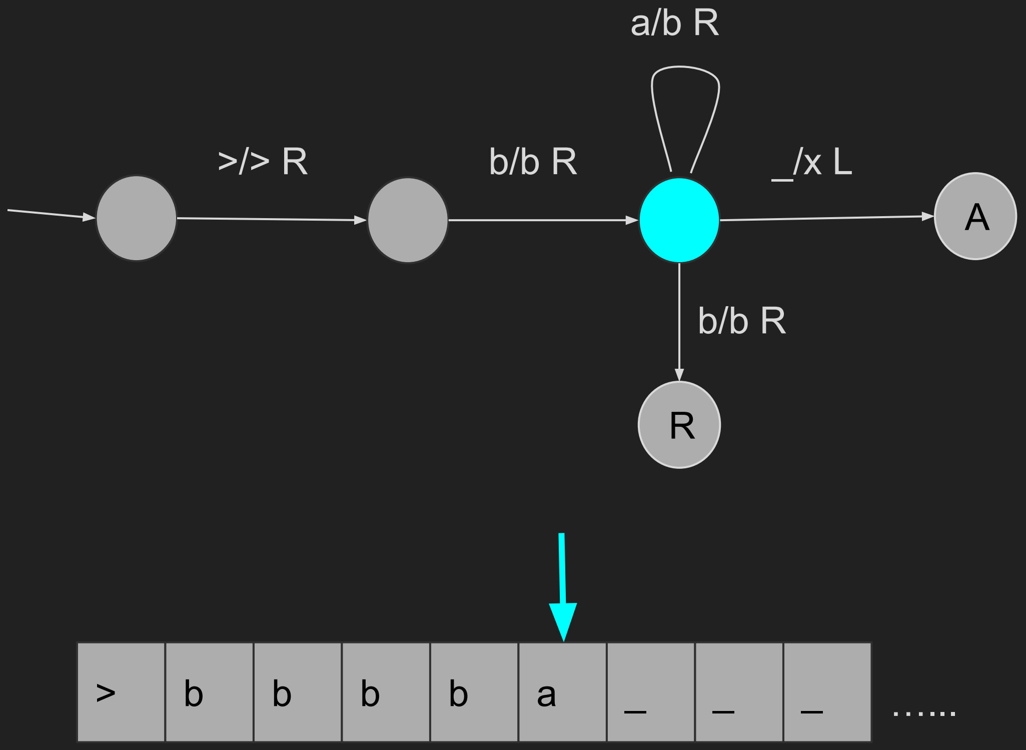
Running the Turing machine



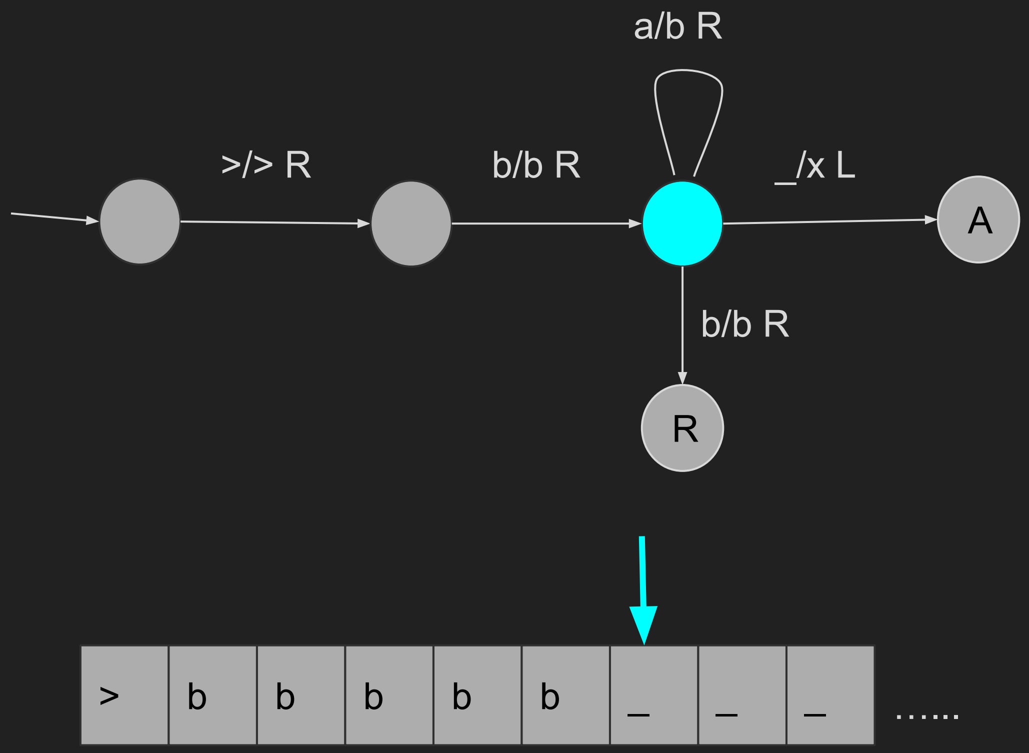
Running the Turing machine



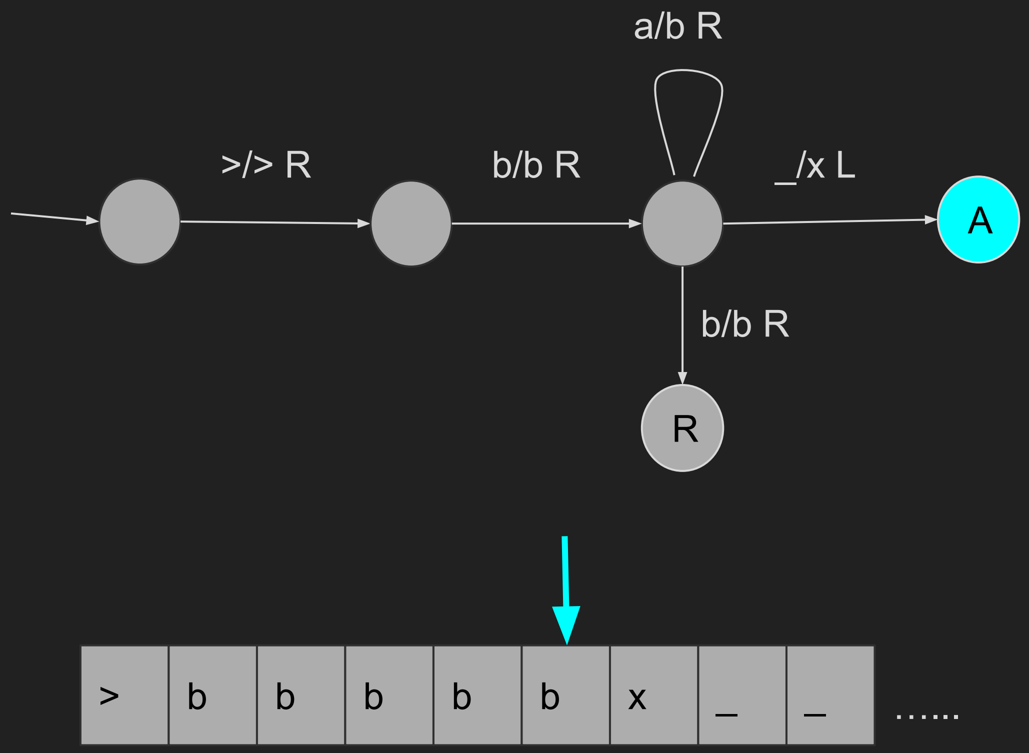
Running the Turing machine



Running the Turing machine



Running the Turing machine

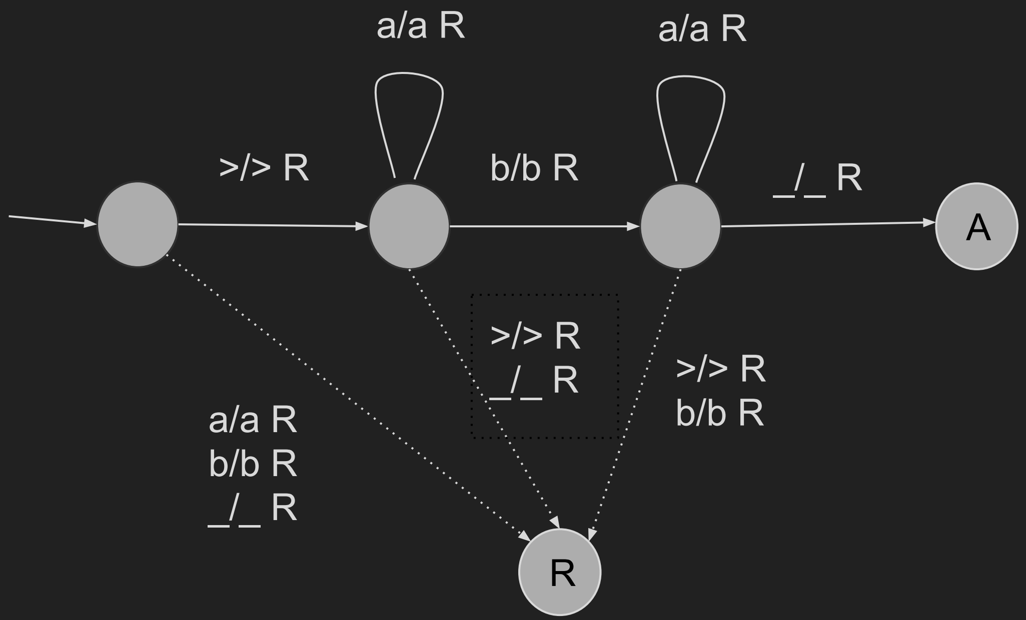


If you ever reach the accept state, ACCEPT the original input string

If you ever reach the reject state, REJECT the original input string

You may never reach the accept or reject state — you keep going forever

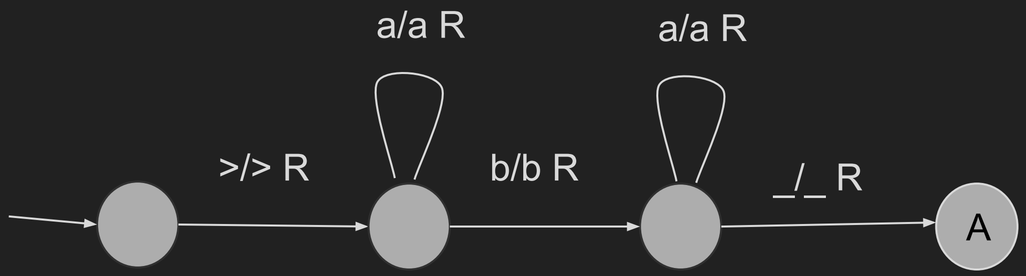
Example — a*ba*



If a language is regular, you can build a TM that accepts all strings in the language by mimicking a deterministic finite state machine that accepts the language

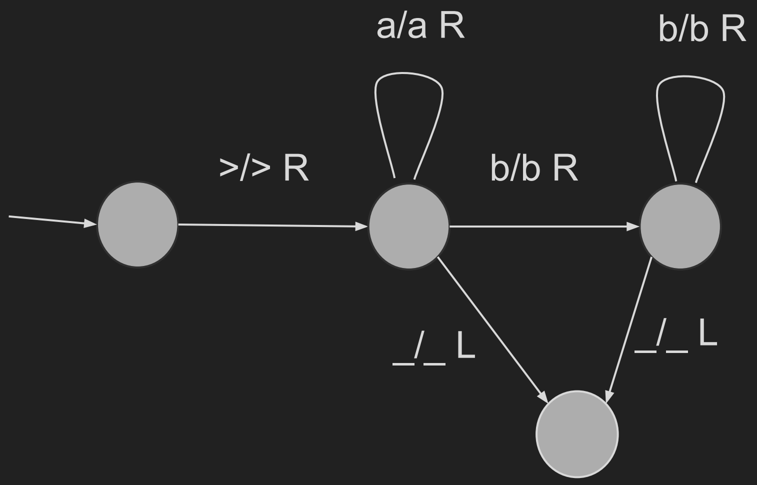
That TM doesn't need to change the tape, and can always move the pointer Right

Example — a*ba*



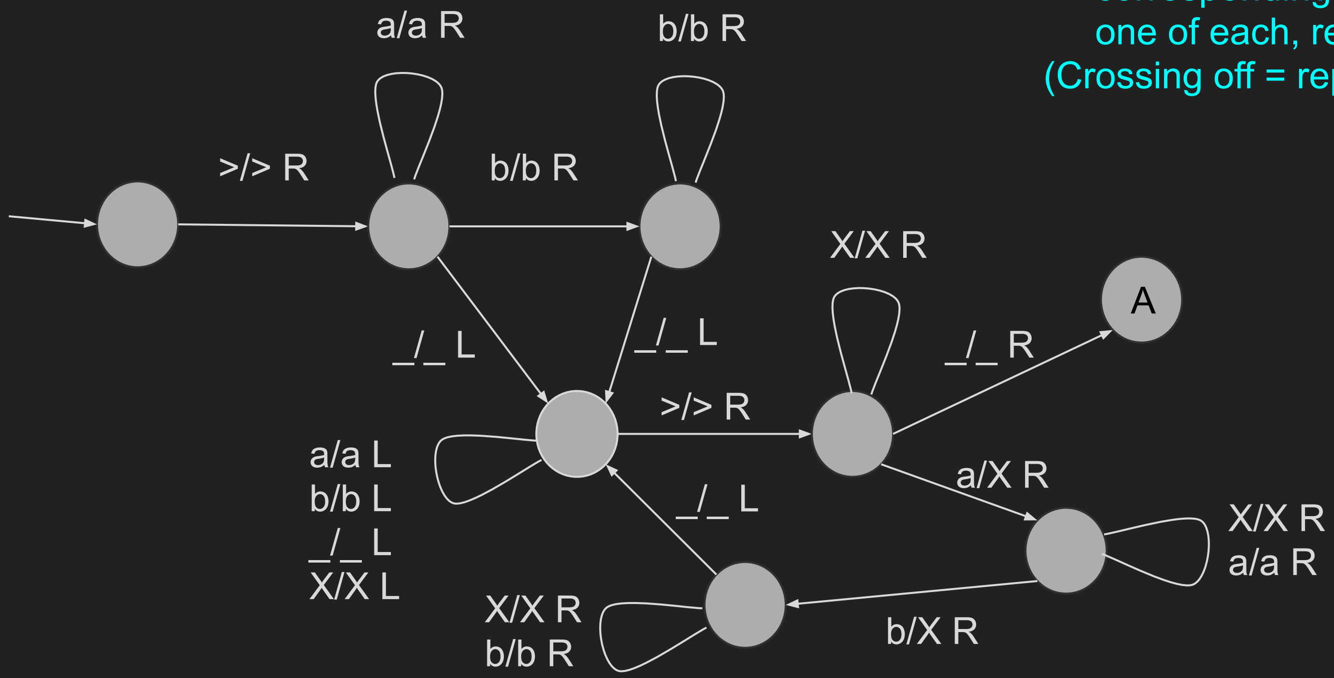
Example — $\{a^n b^n \mid n \geq 0\}$

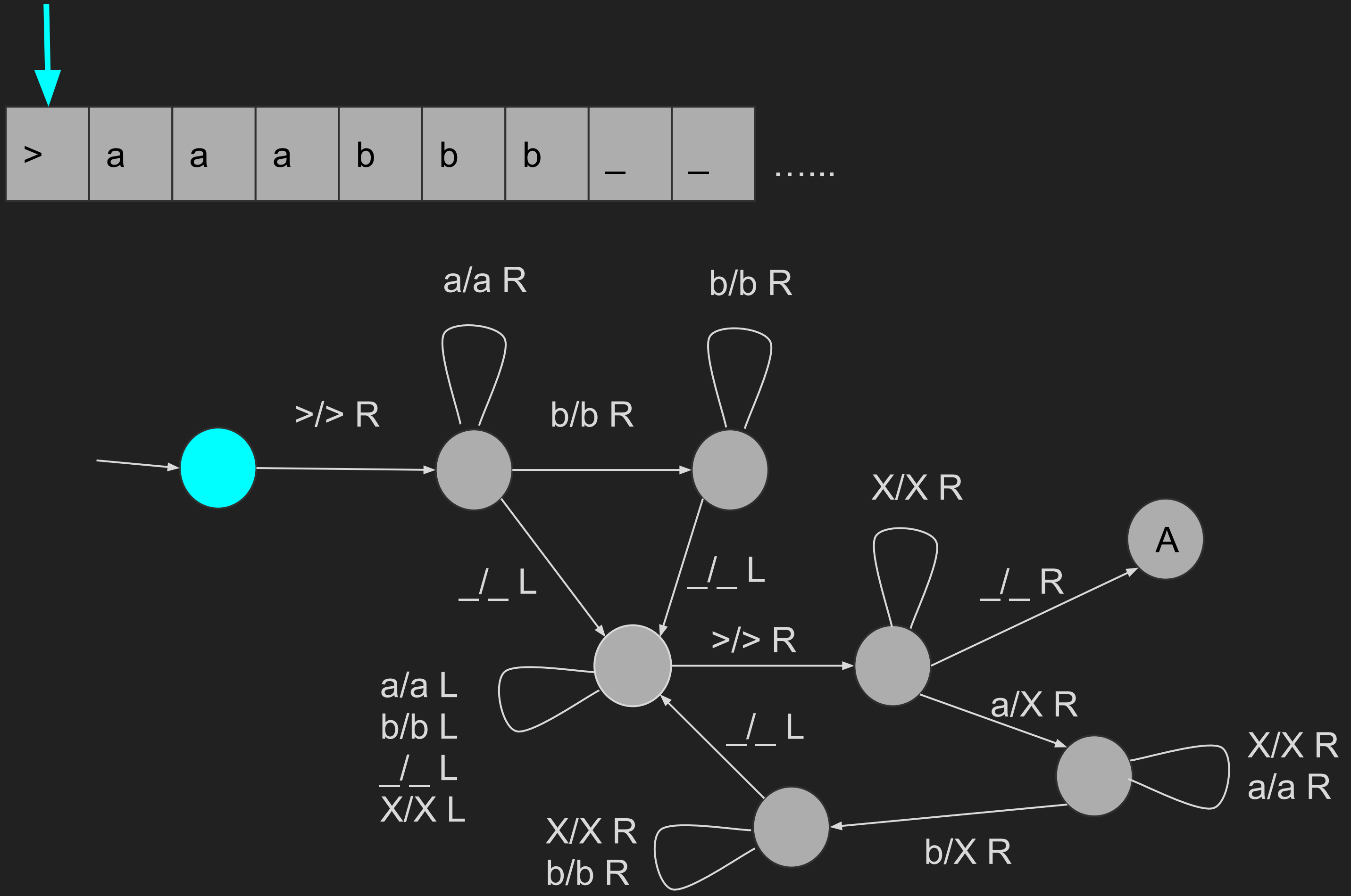
Intuition:
(1) check that we have a*b*



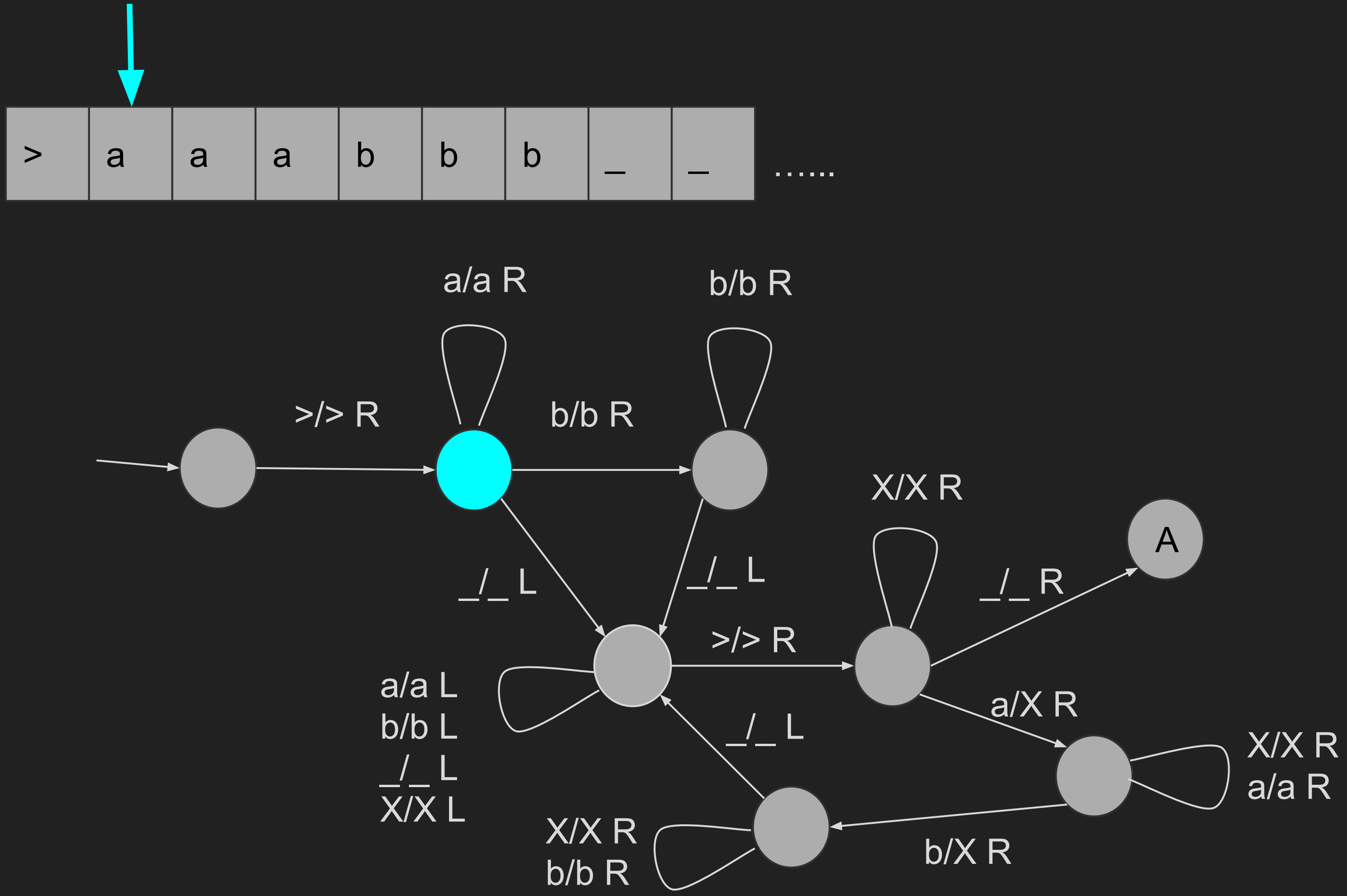
Example — $\{a^n b^n \mid n \geq 0\}$

Intuition:
(1) check that we have a*b*
(2) check that every a has a corresponding b by crossing off one of each, repeatedly
(Crossing off = replace by X)

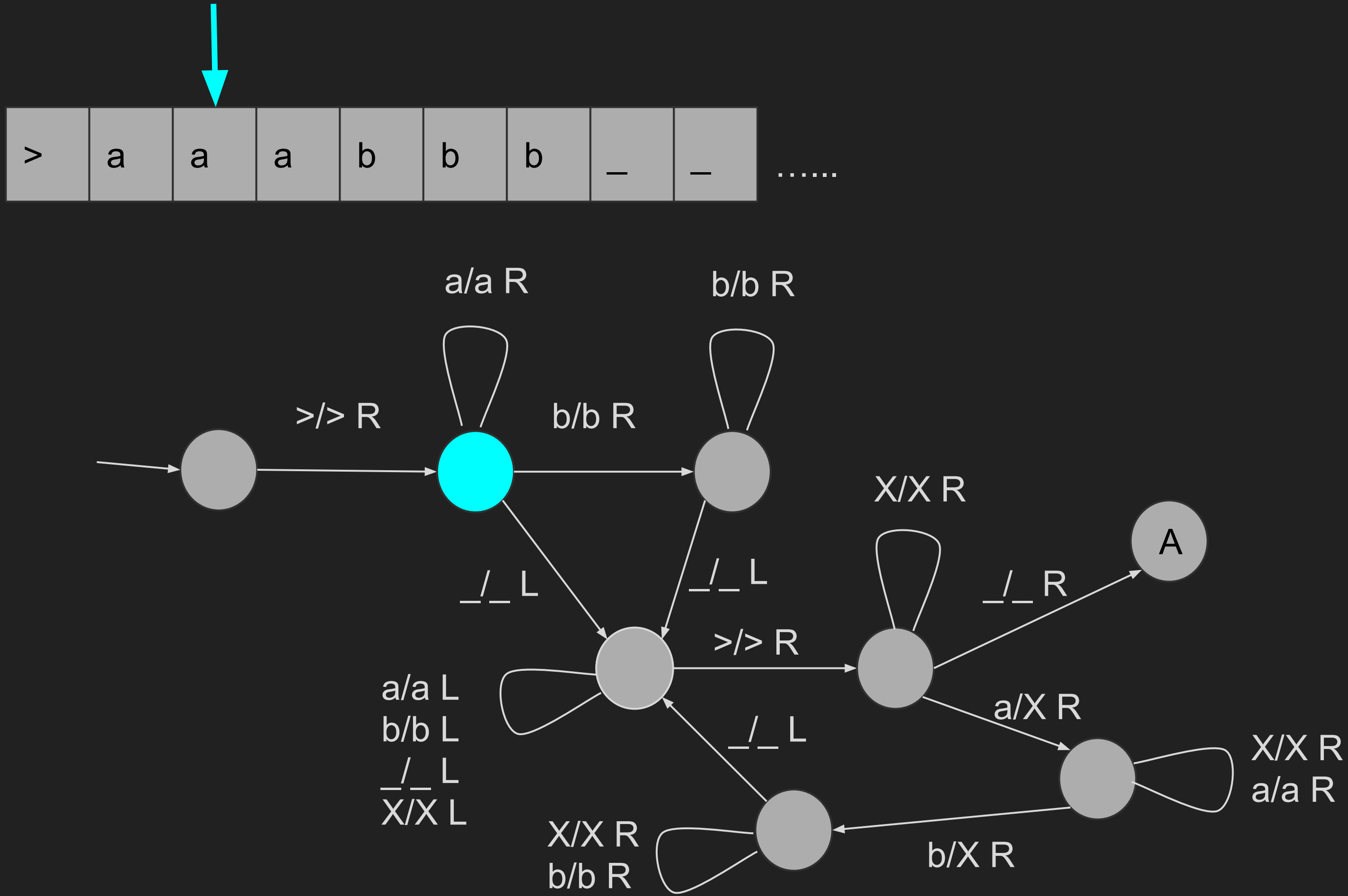


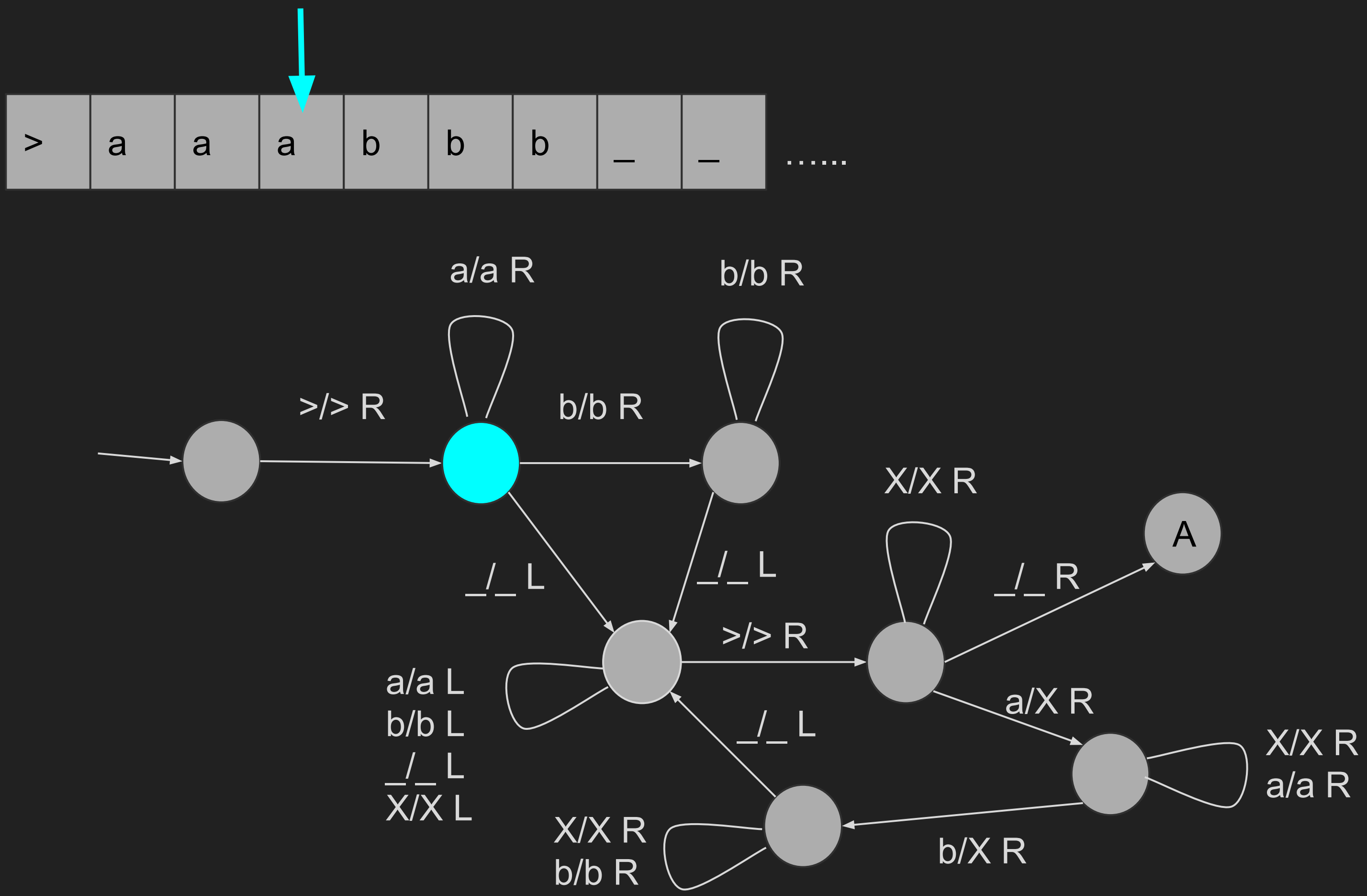


>
a
a
a
b
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

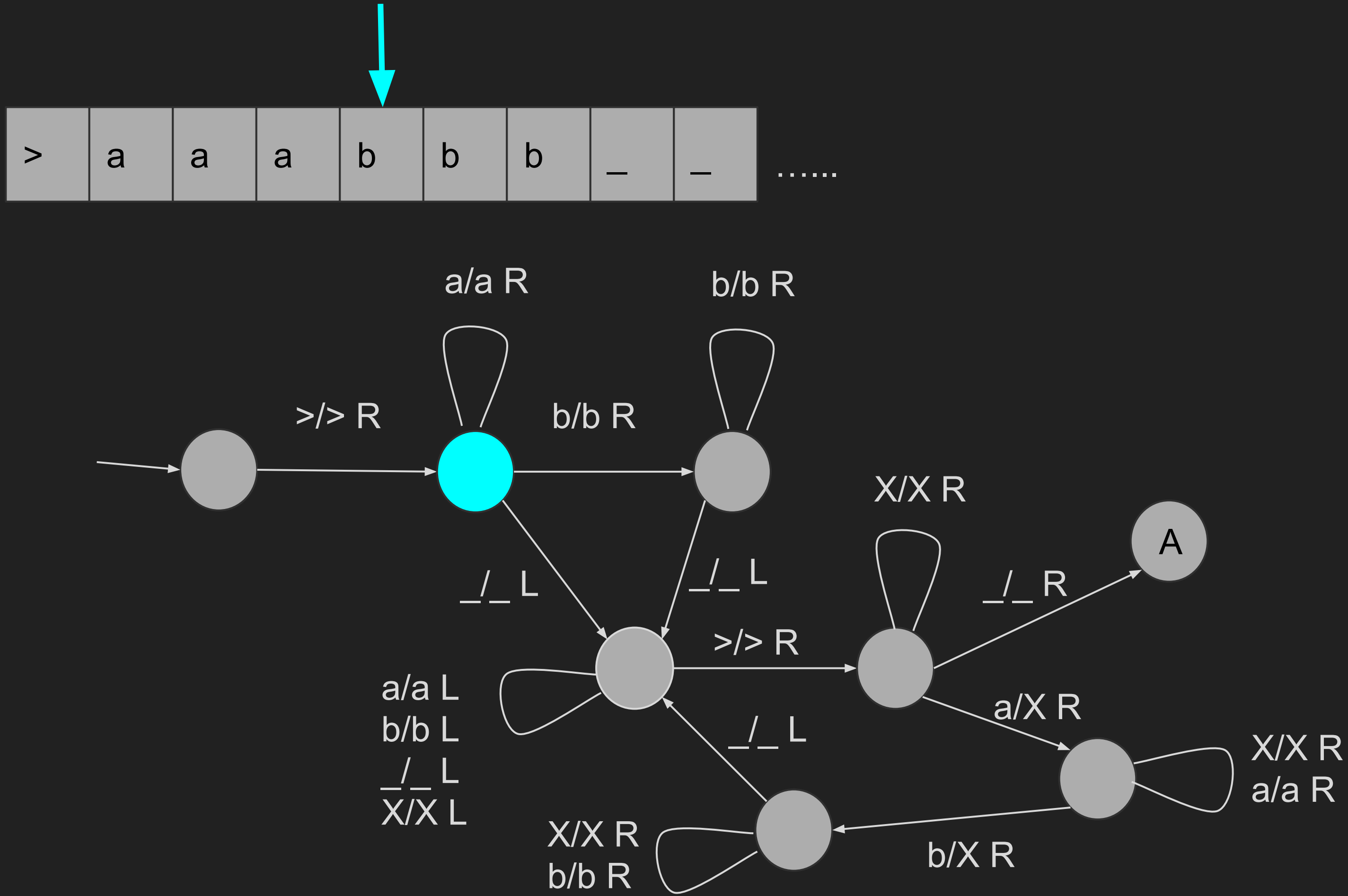


>
a
a
a
b
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

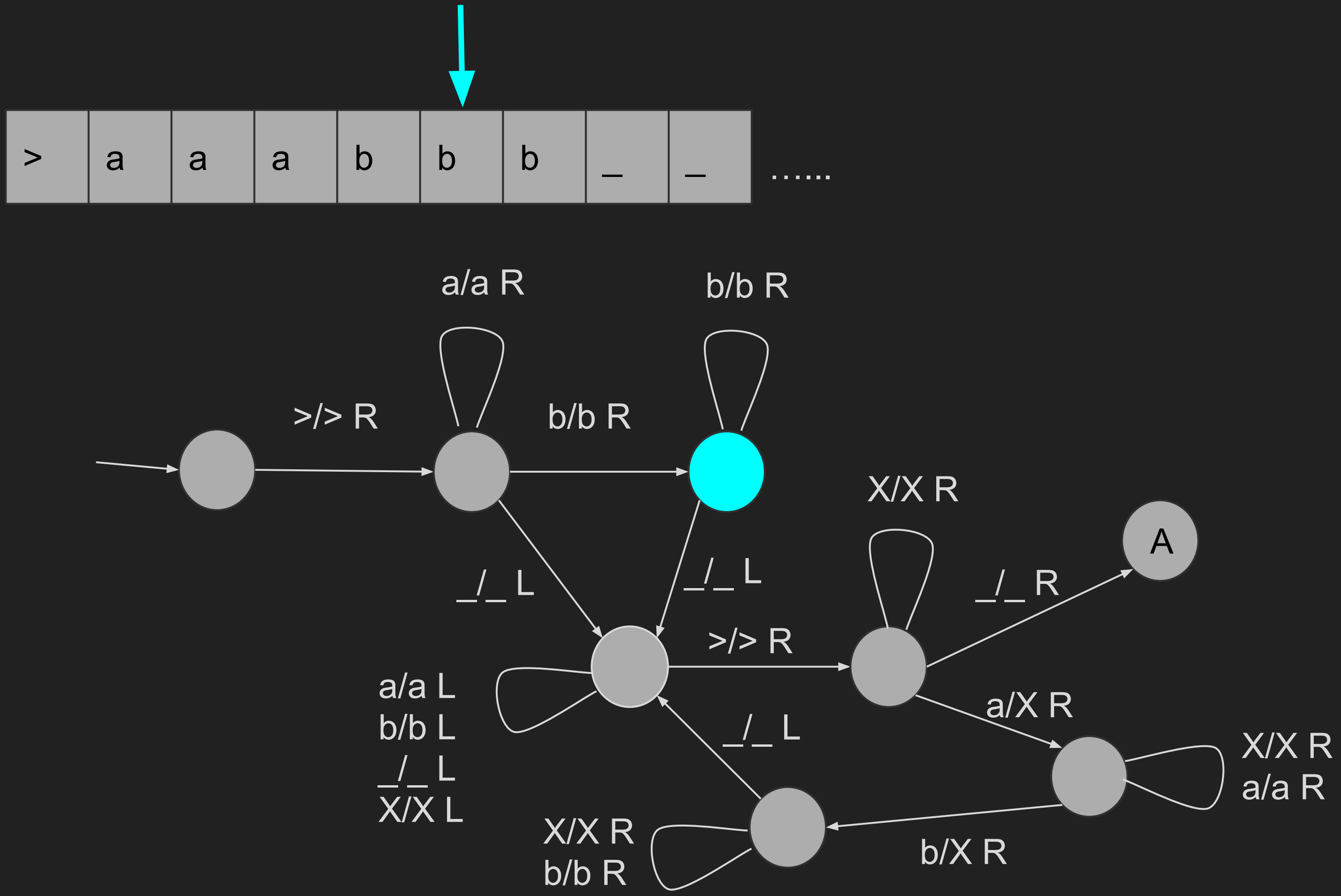

>
a
a
a
b
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



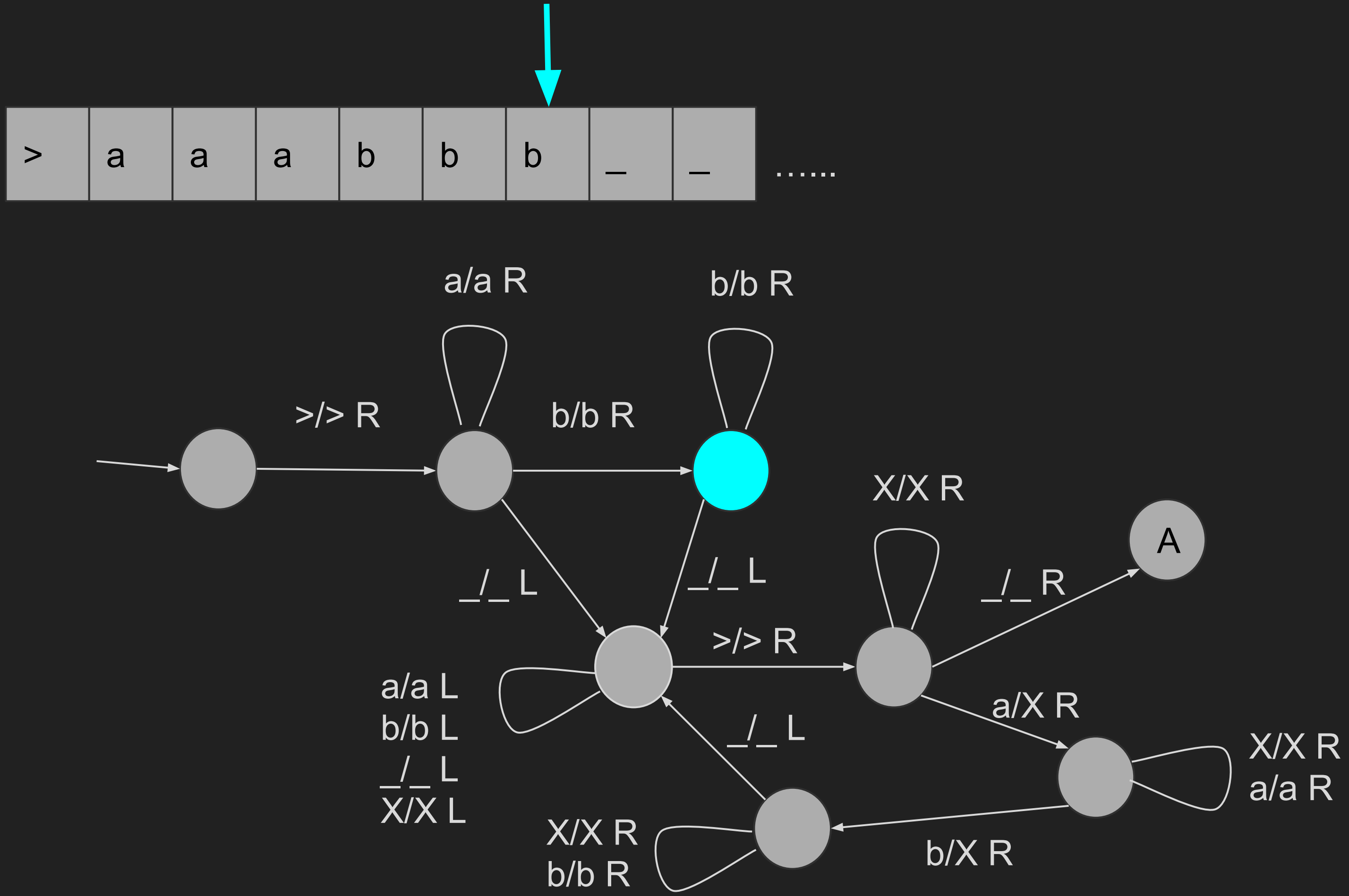
>
a
a
a
b
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

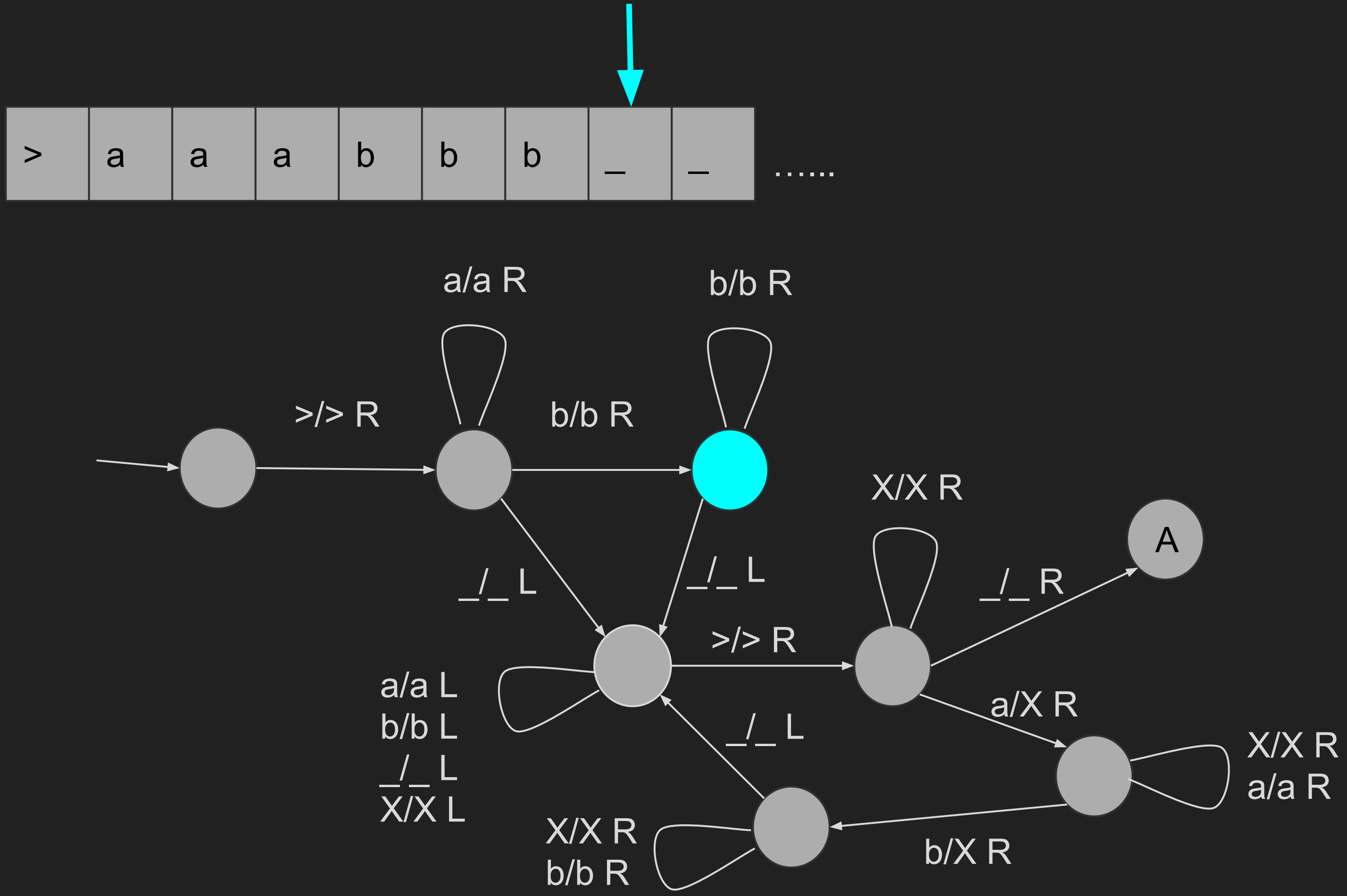


> a a a b b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

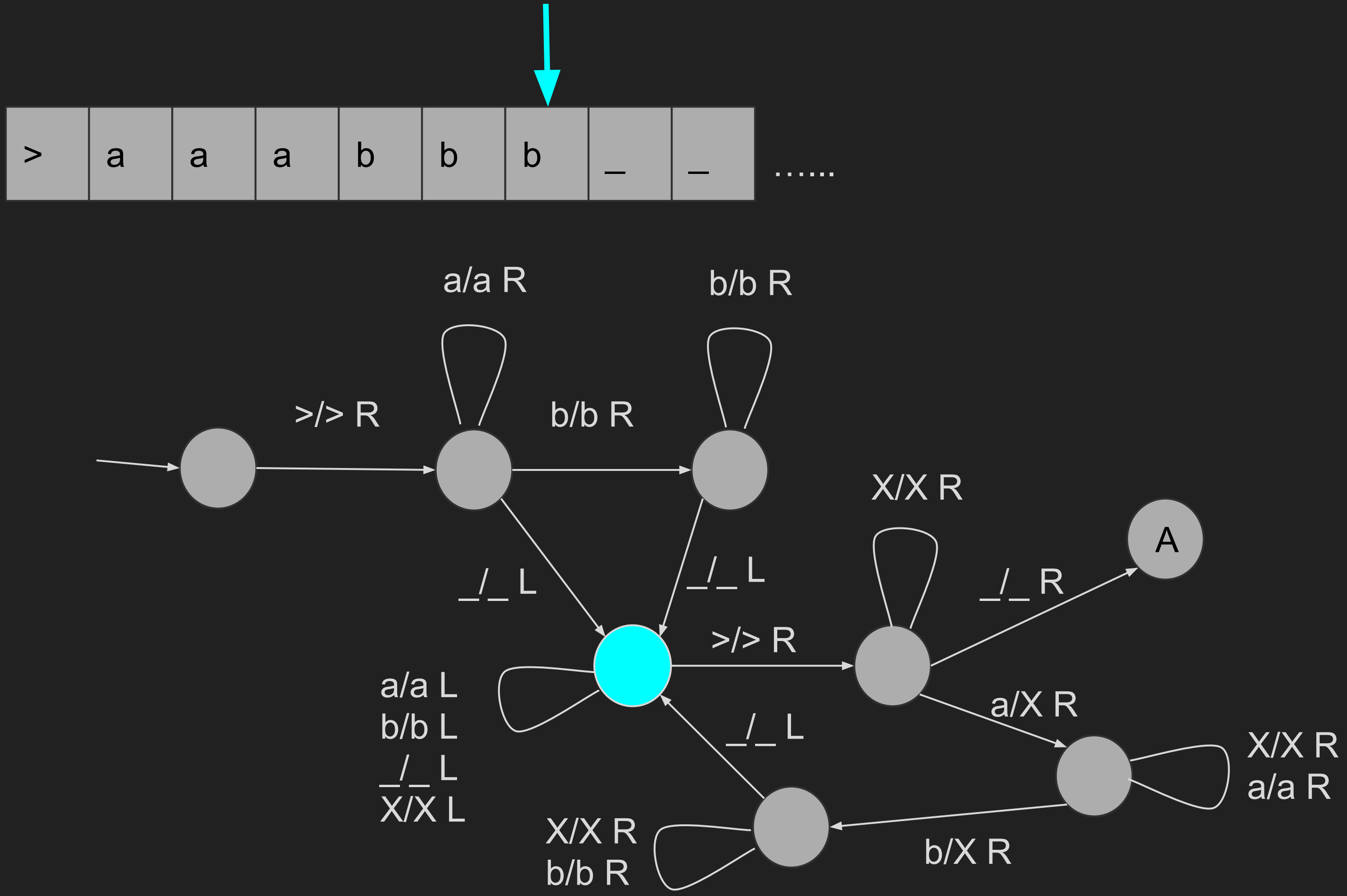


> a a a b b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
>/> R
_/_ R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

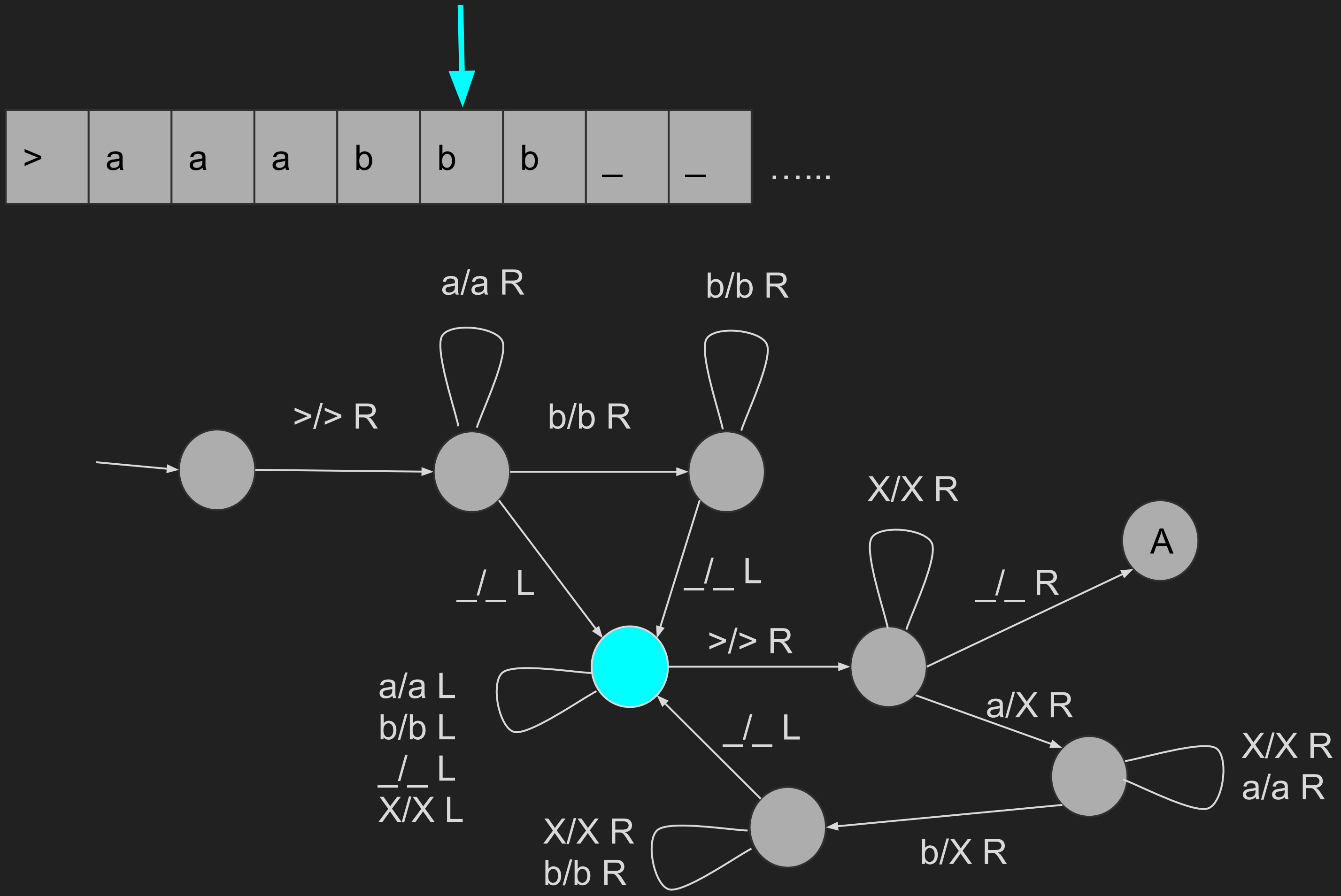

> a a a b b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



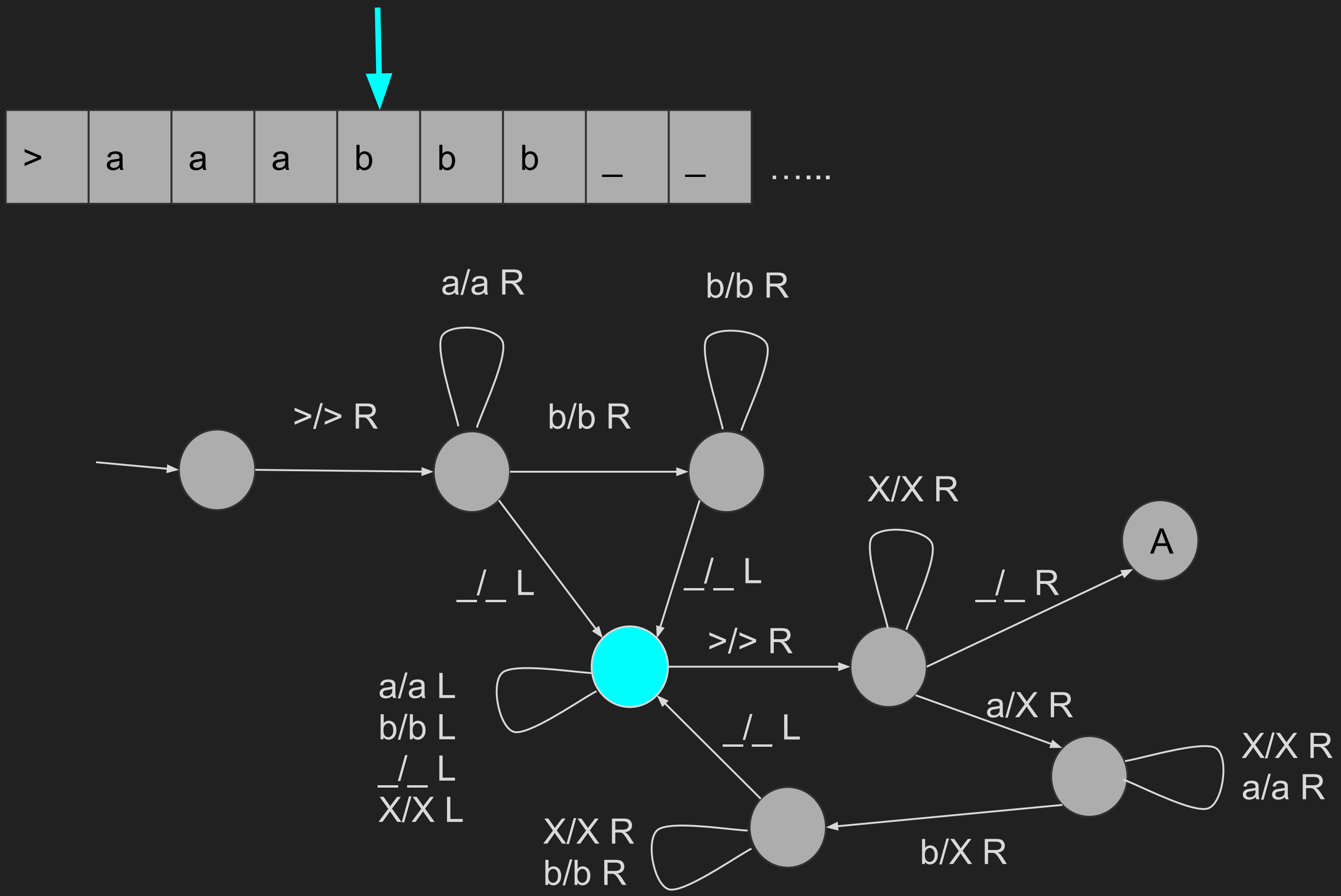
>
a
a
a
b
b
b
_
_
......
>/> R
a/a R
b/b R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



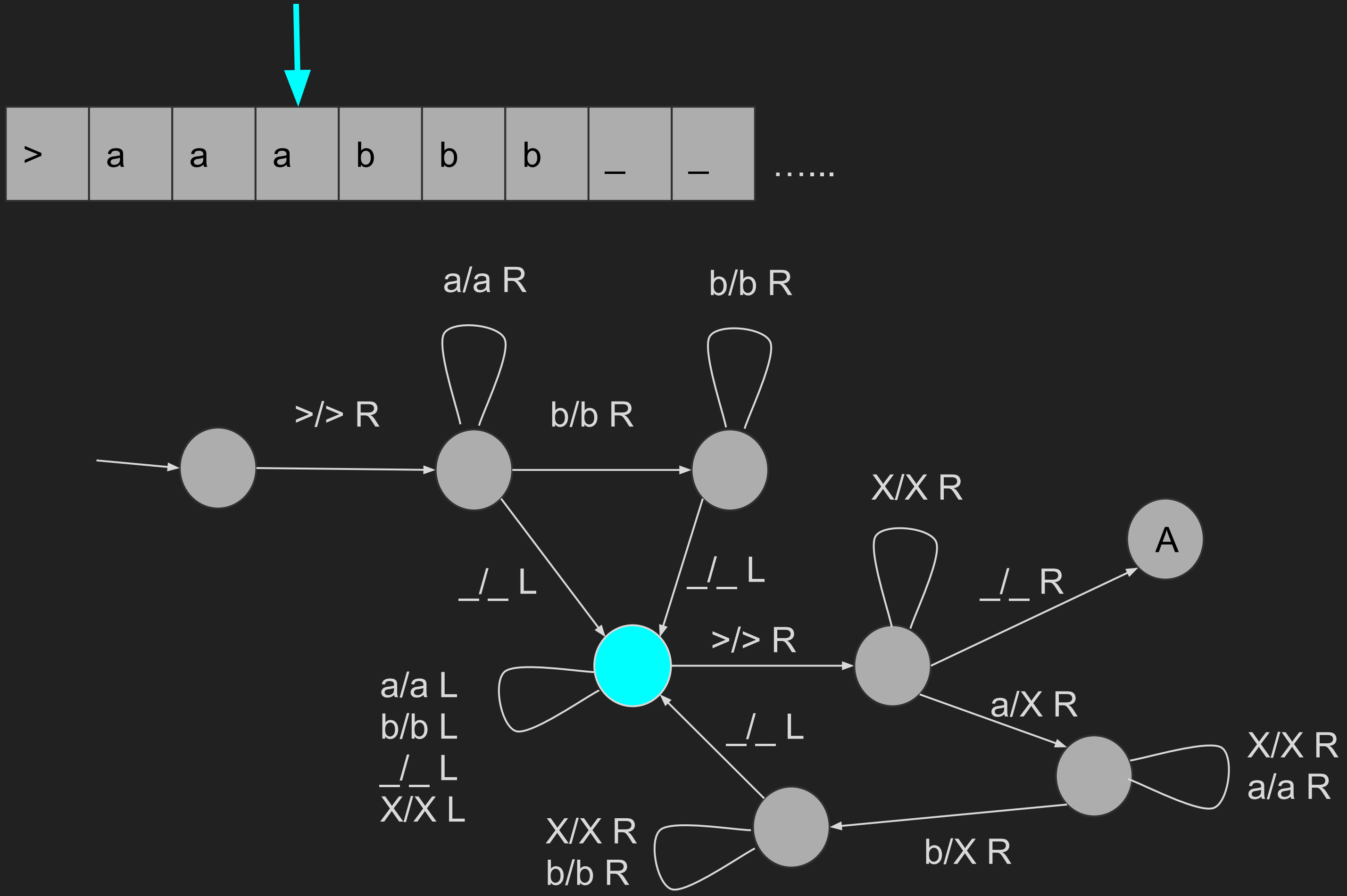
>
a
a
a
b
b
b
_
_
……
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



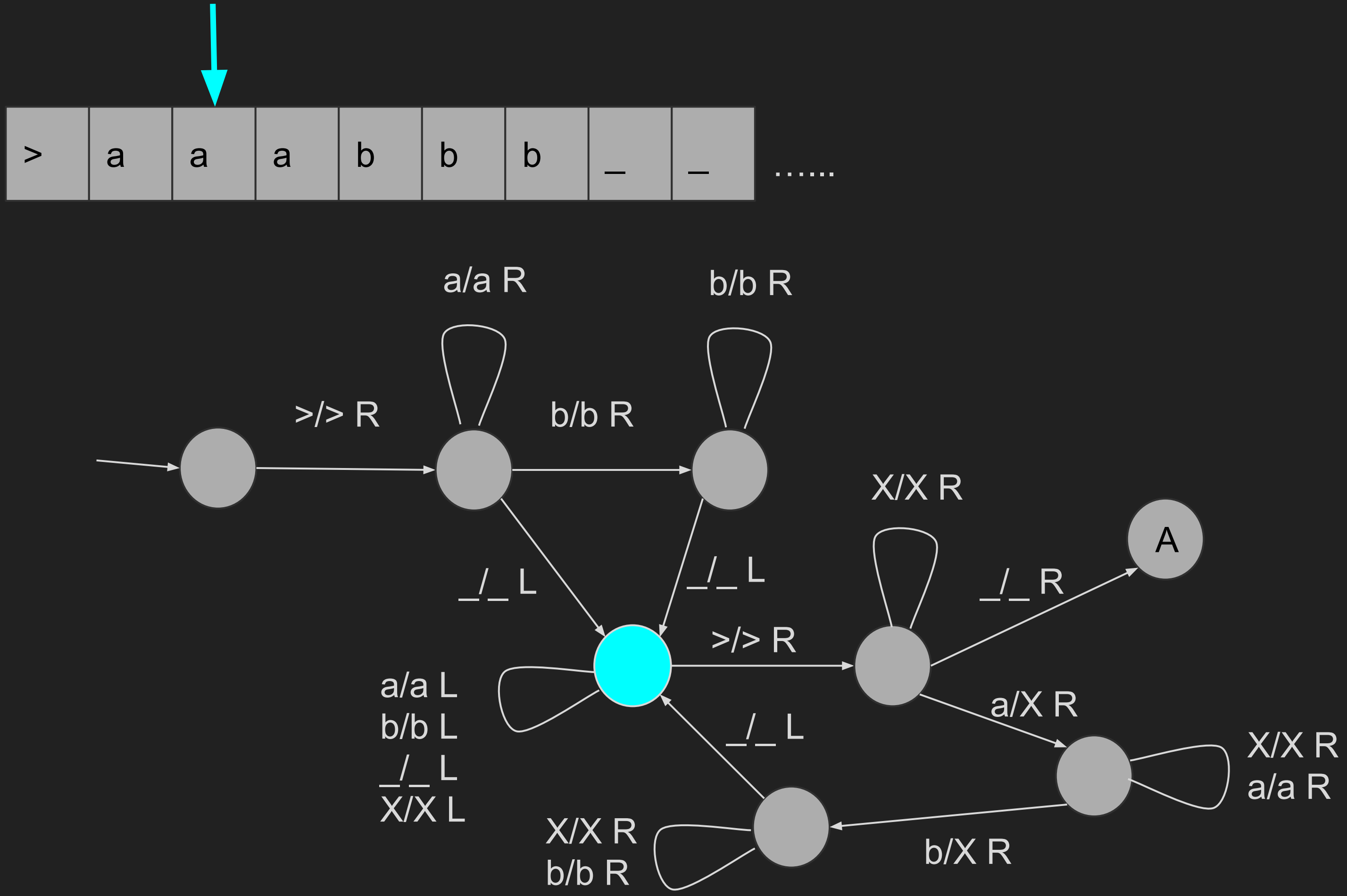
> a a a b b b _ _ ……
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
X/X R
a/a R
_/_ L
X/X R
b/b R
b/X R



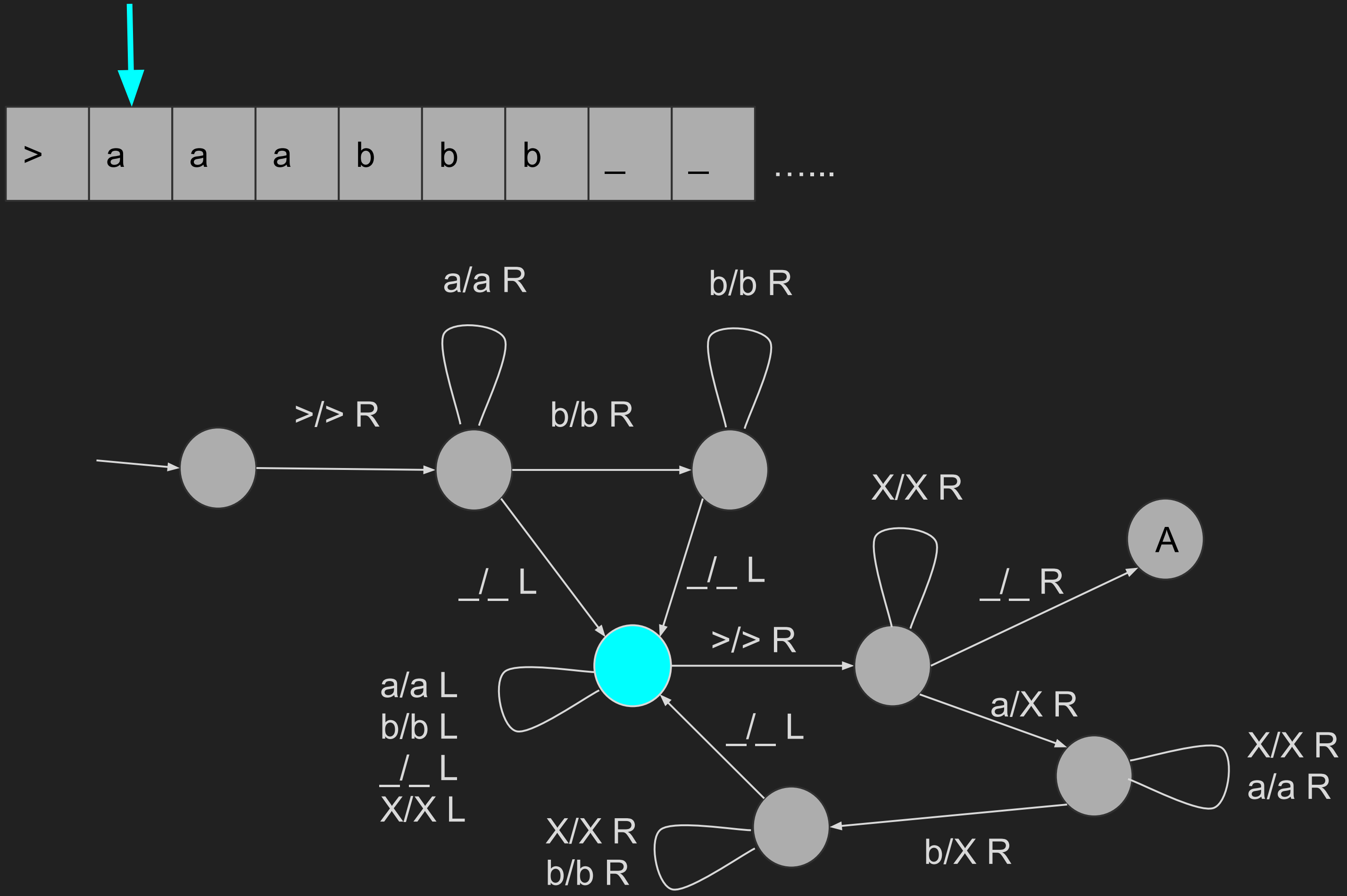
> a a a b b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



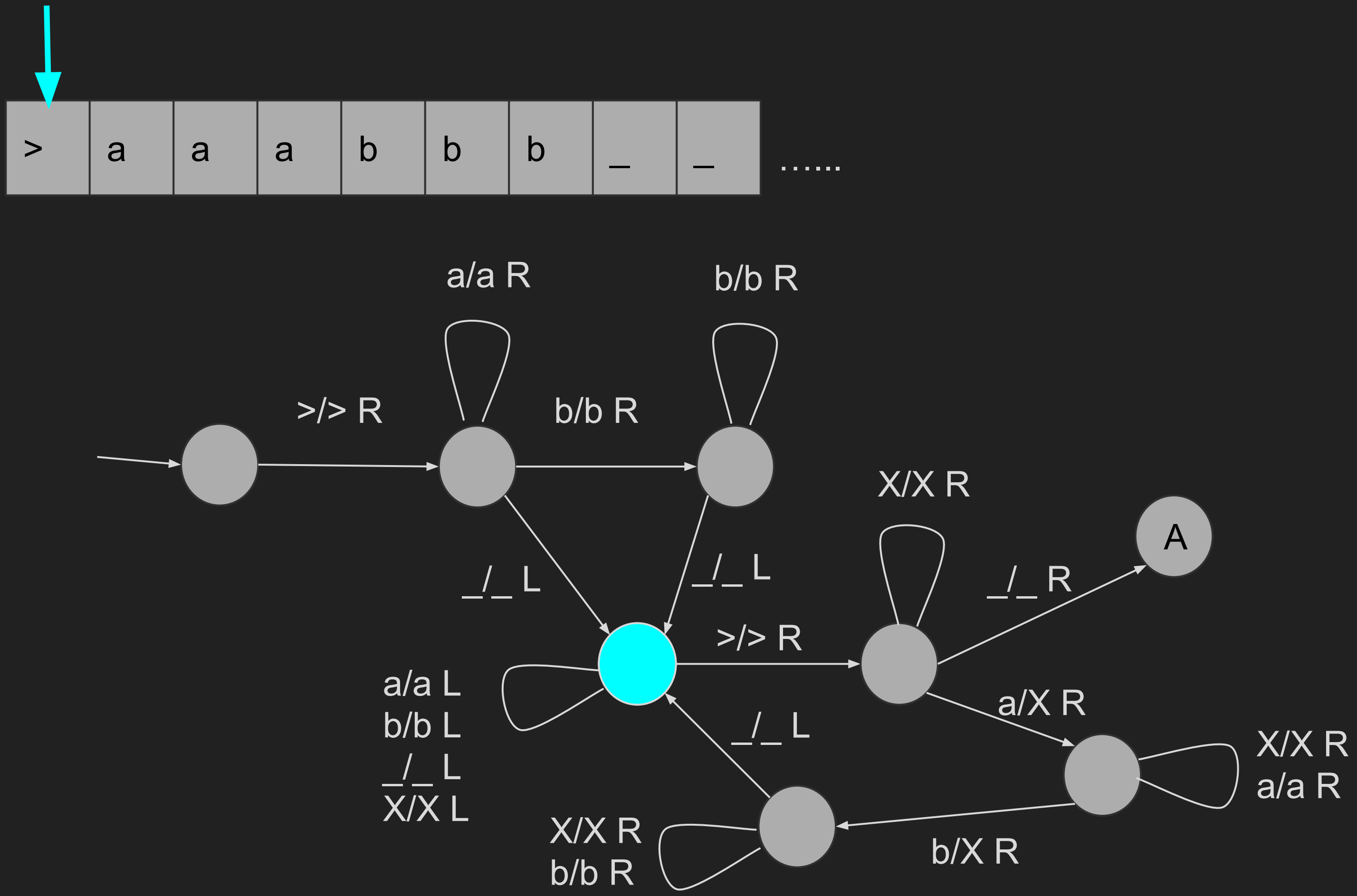
>
a
a
a
b
b
b
_
_
……
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



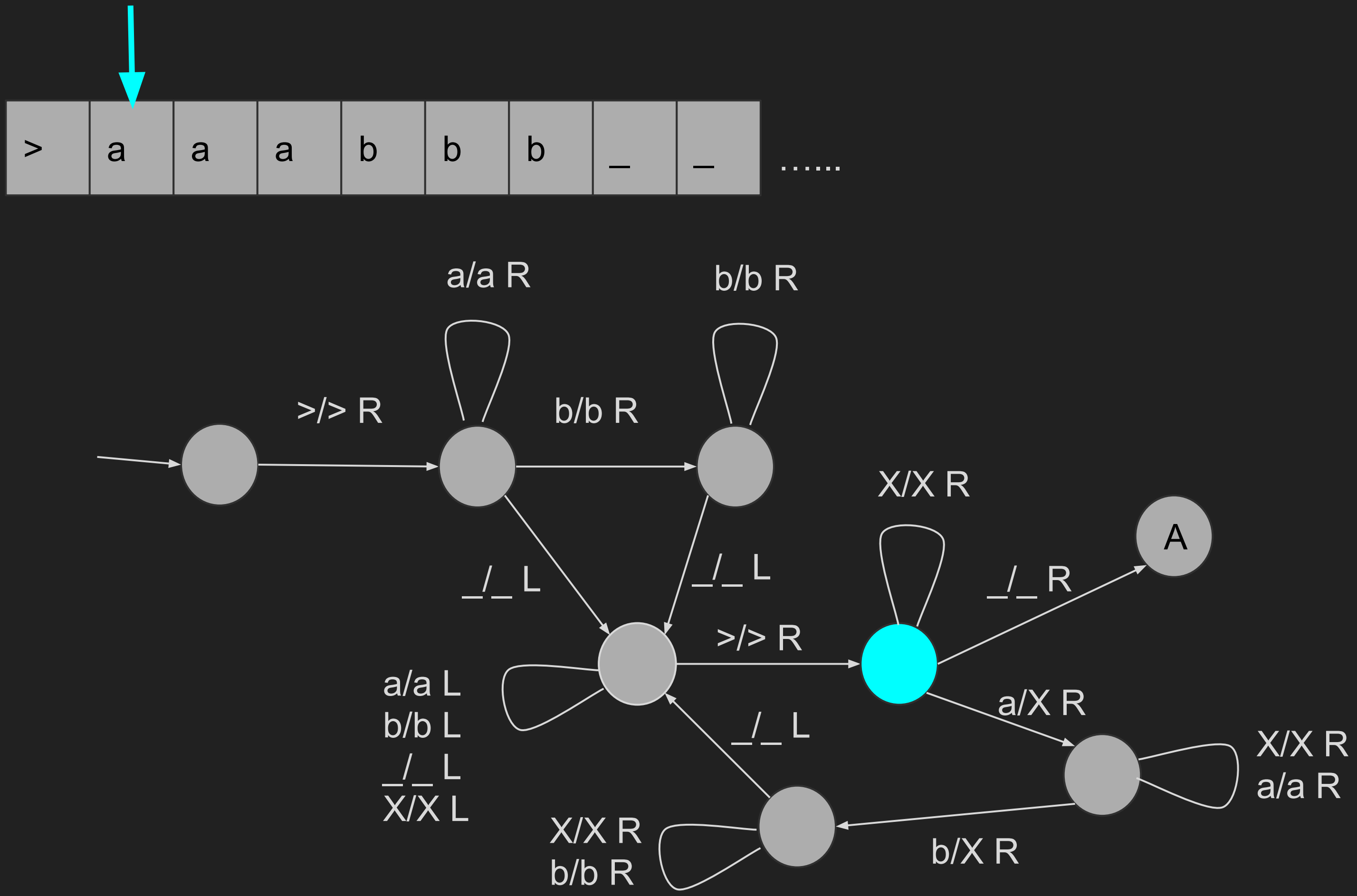
>
a
a
a
b
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



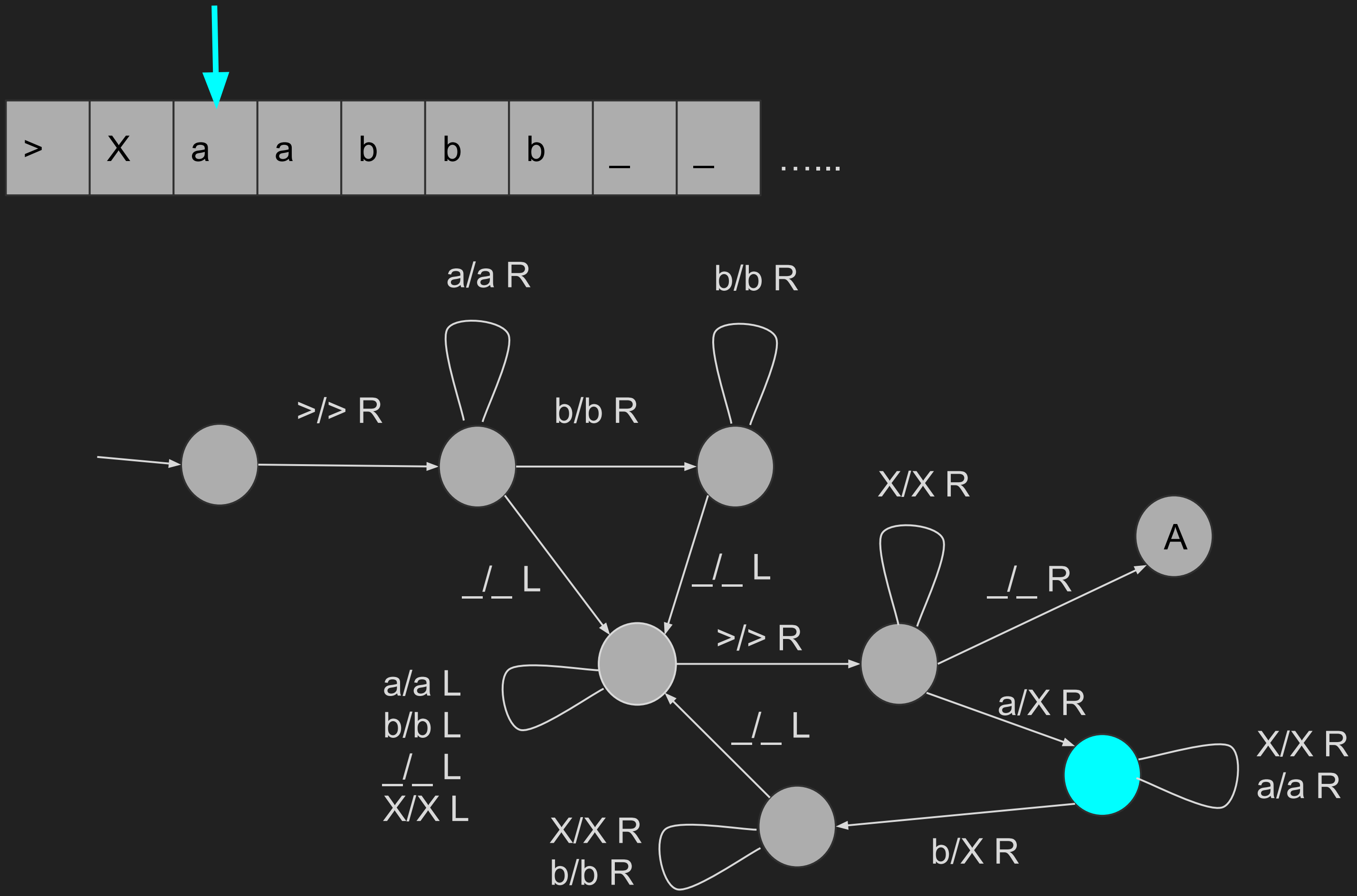
>
a
a
a
b
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



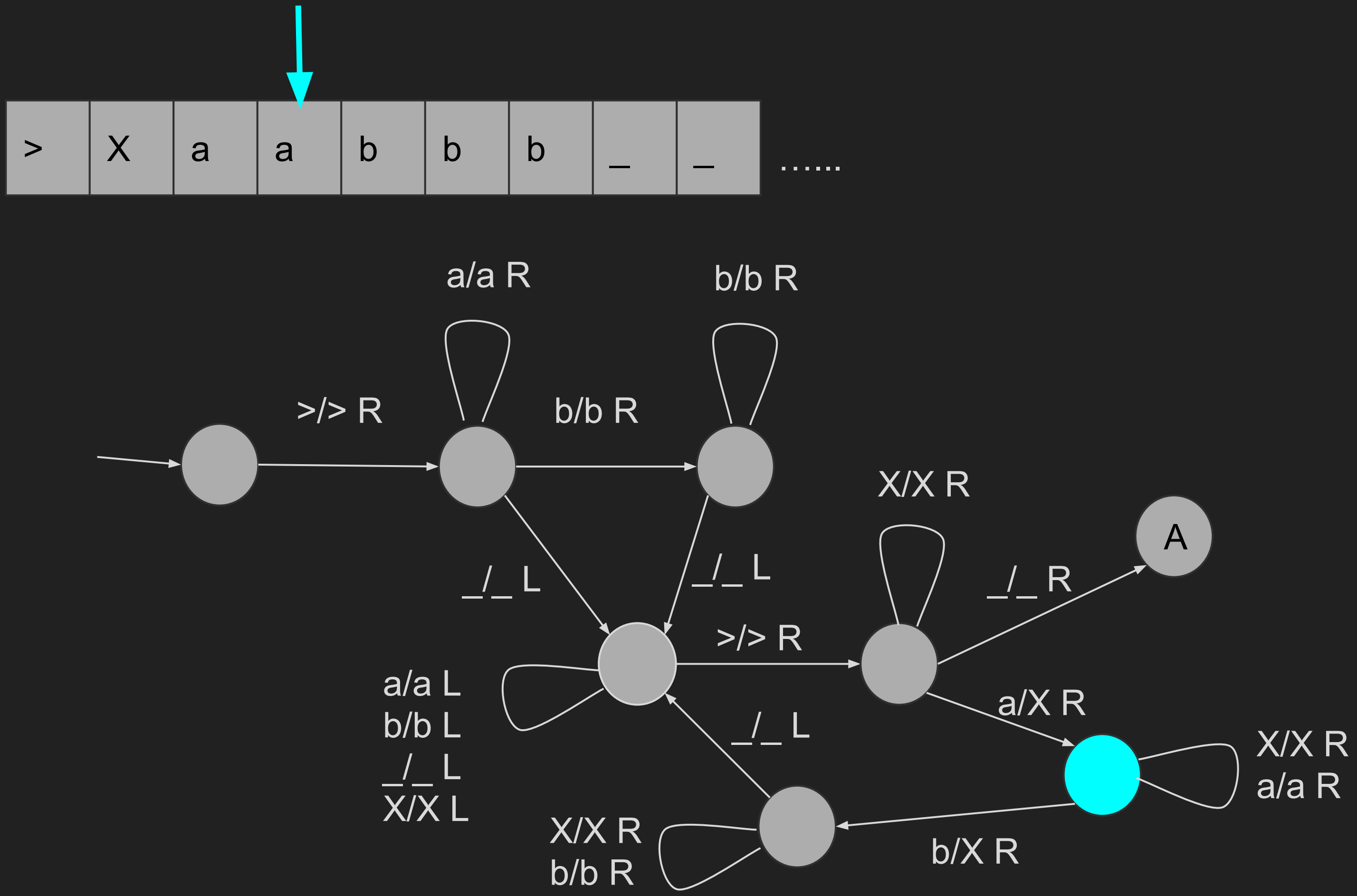
> a a a b b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



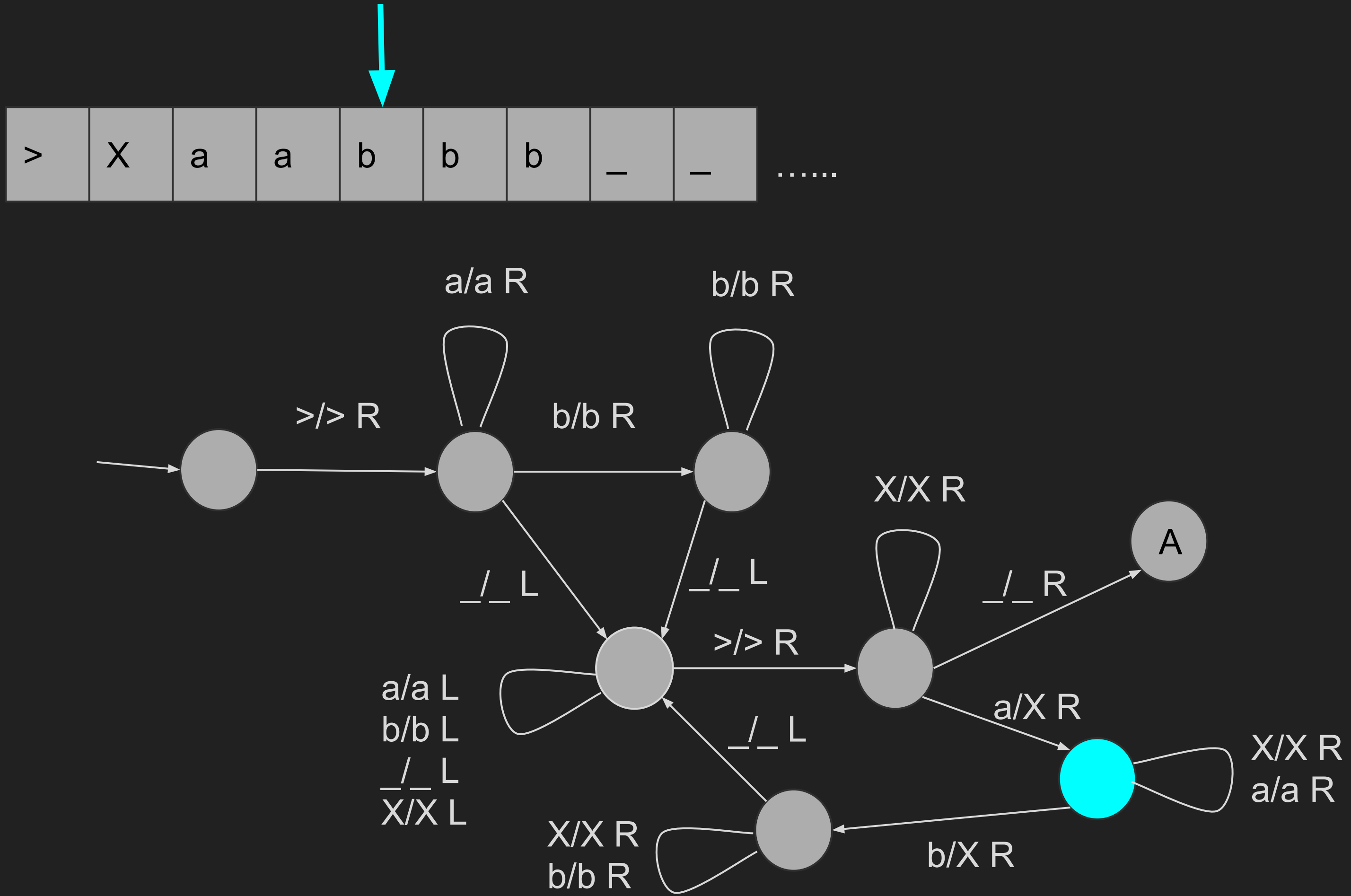
> a a a b b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



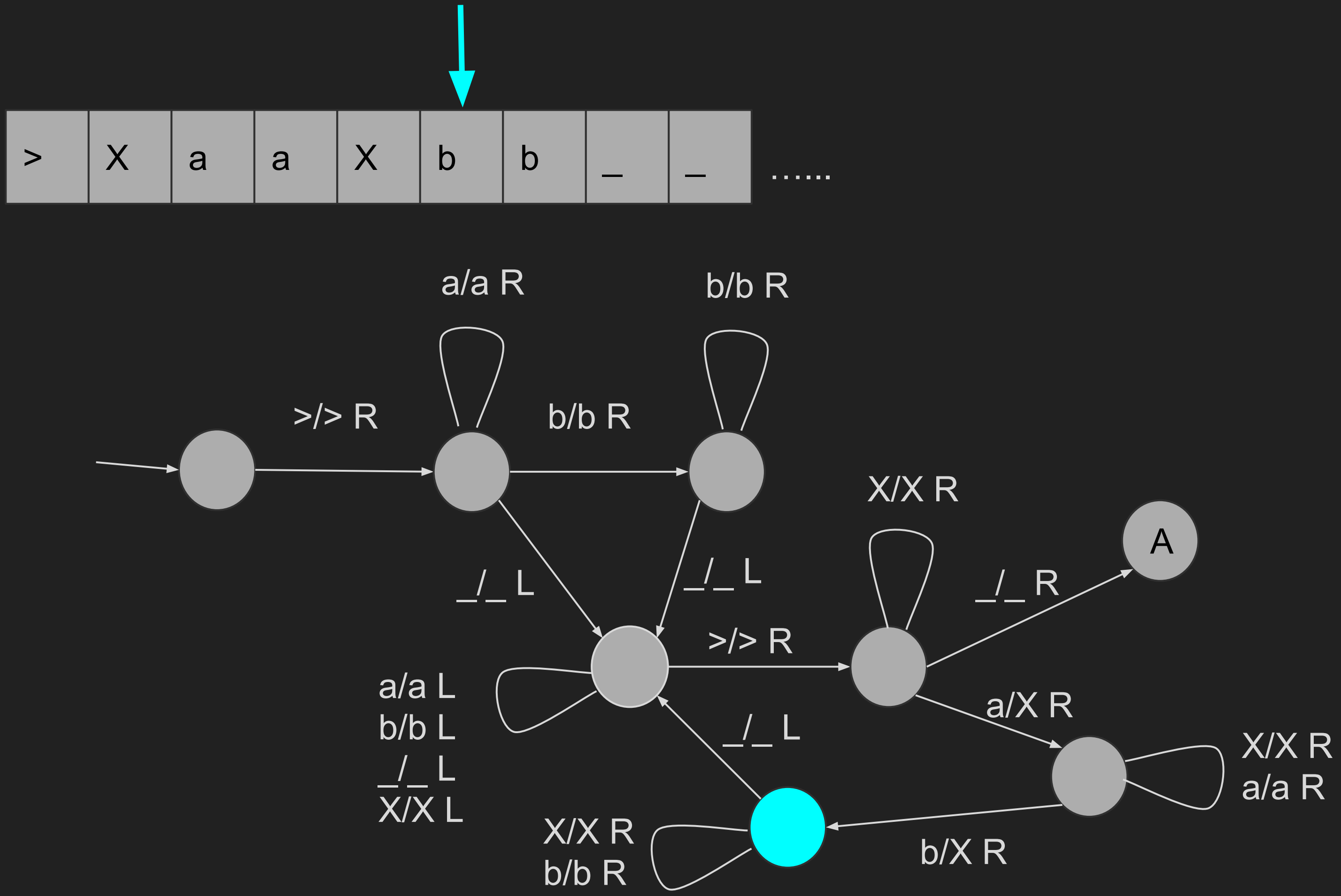
>
X
a
a
b
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



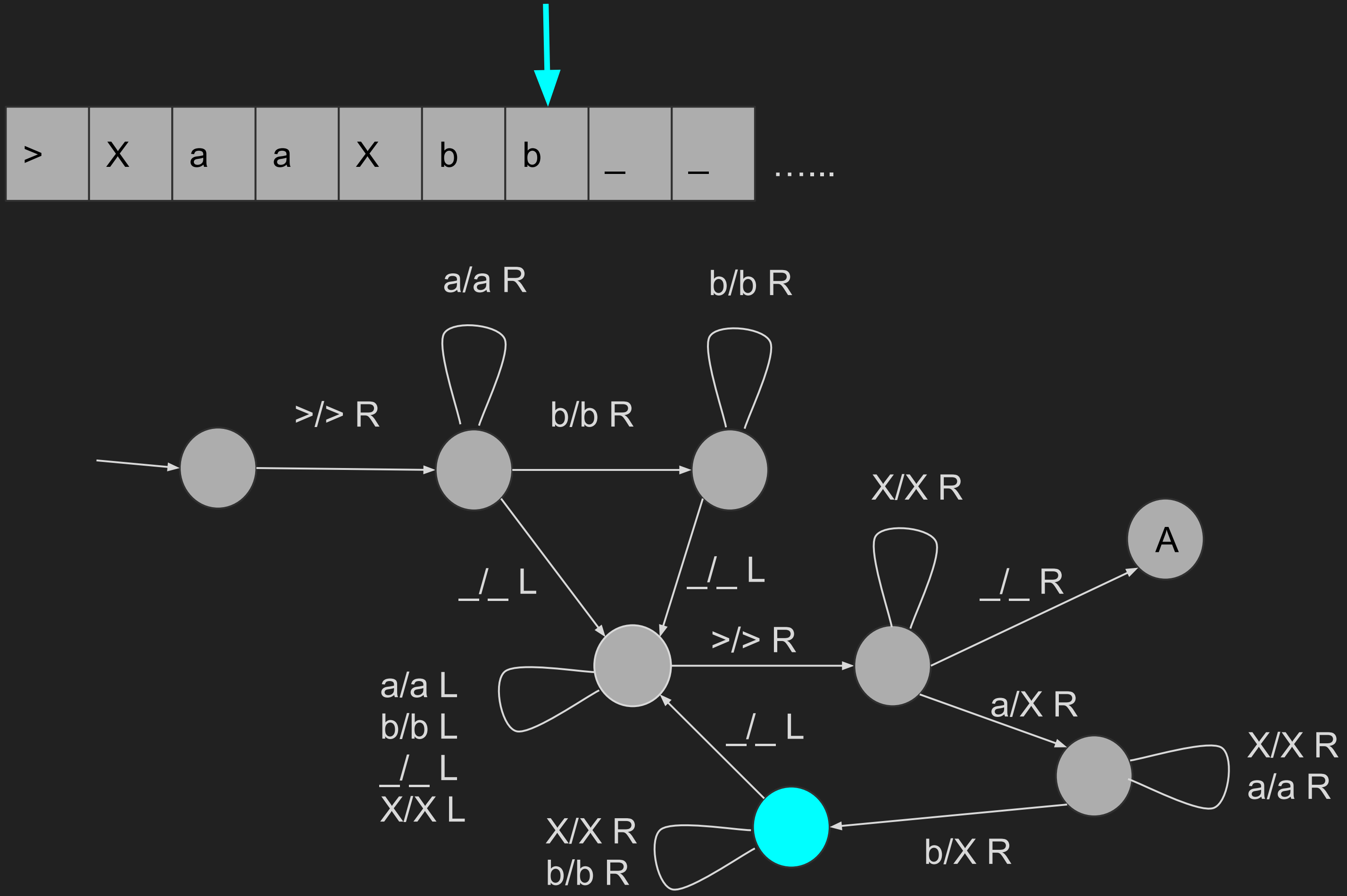
> X a a b b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



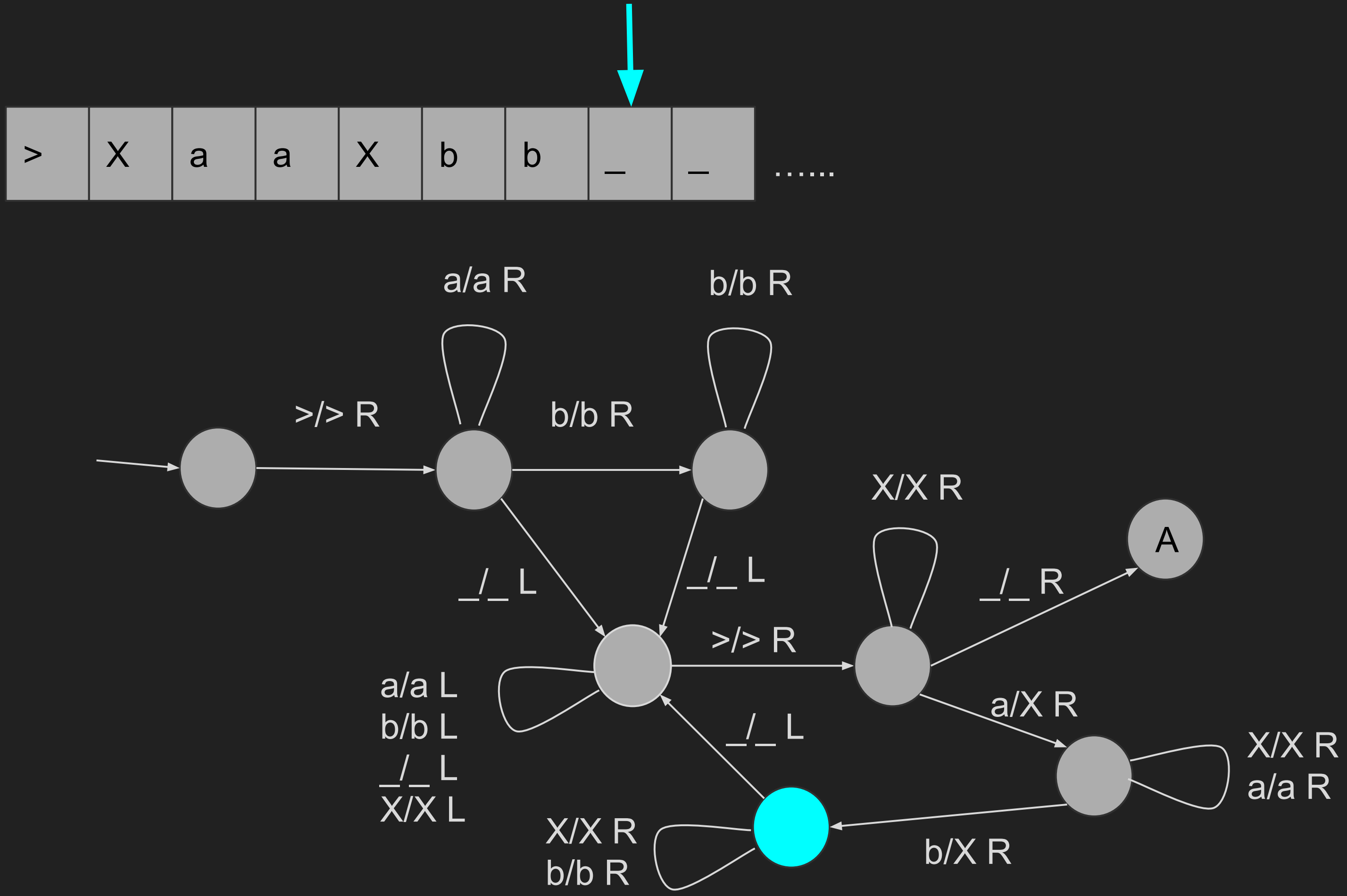
>
X
a
a
b
b
b
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
_/_ L
a/X R
X/X R
a/a R
X/X R
b/b R
b/X R



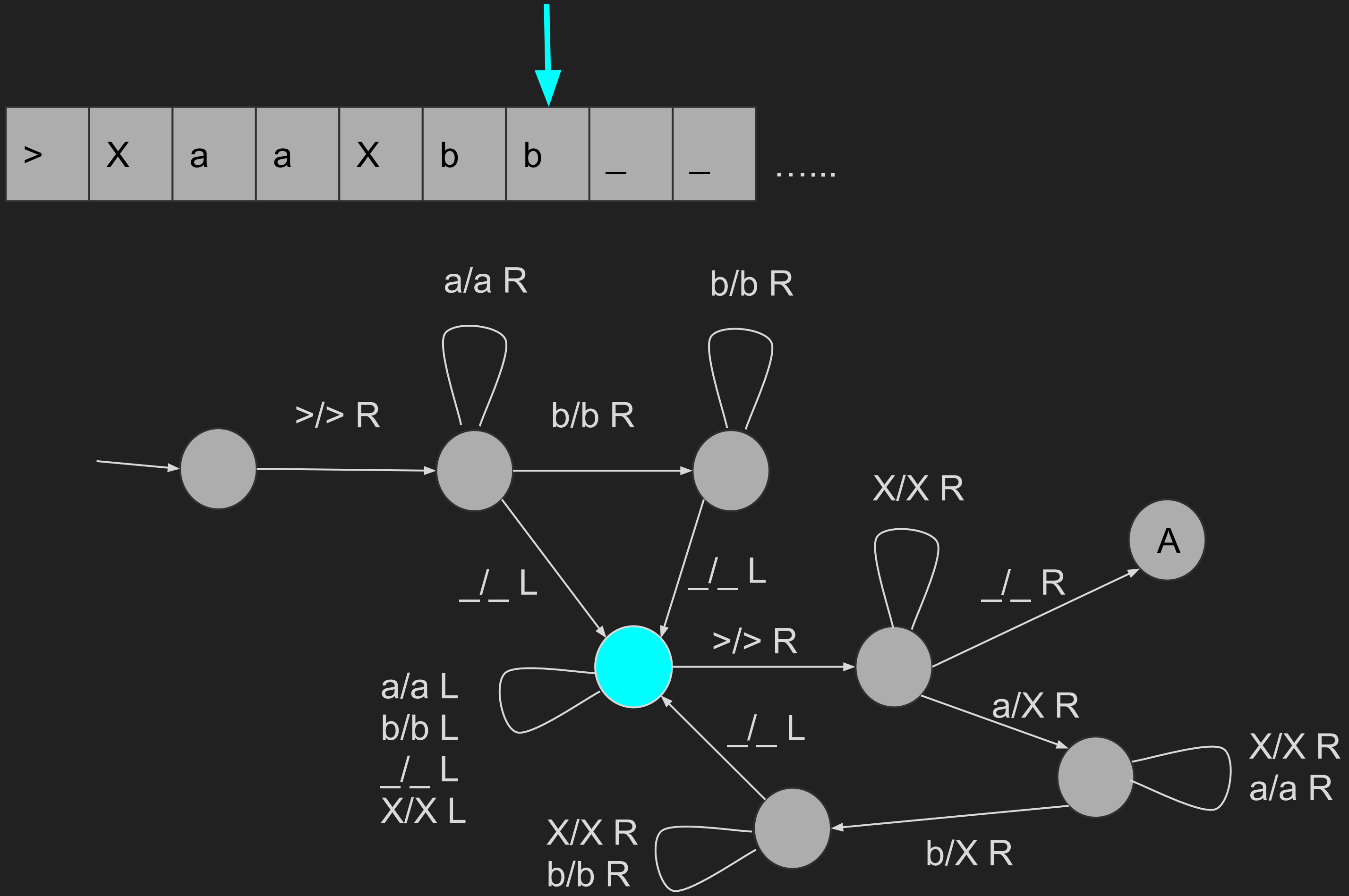
>
X
a
a
X
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



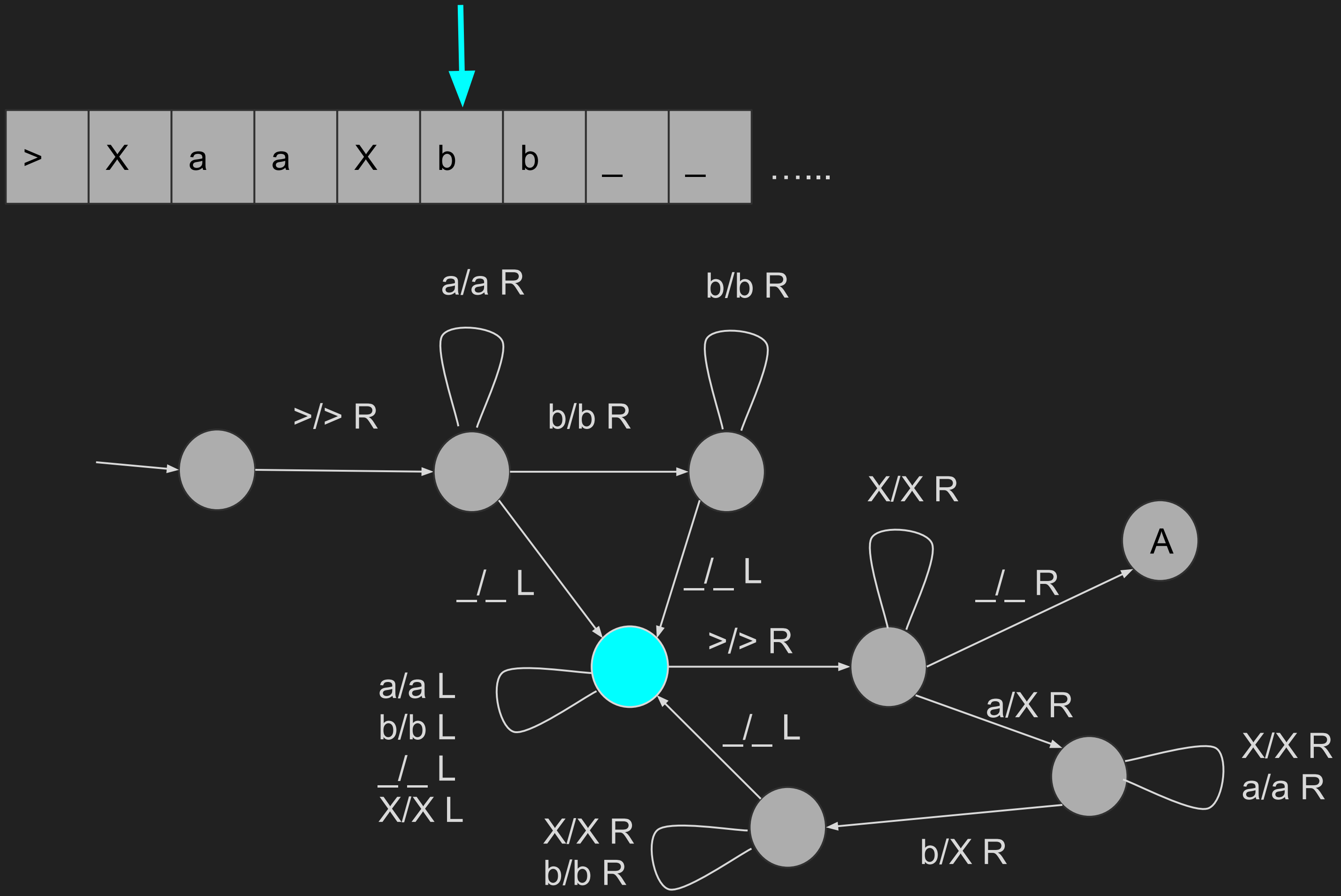
> X a a X b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
>/> R
_/_ R
A
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



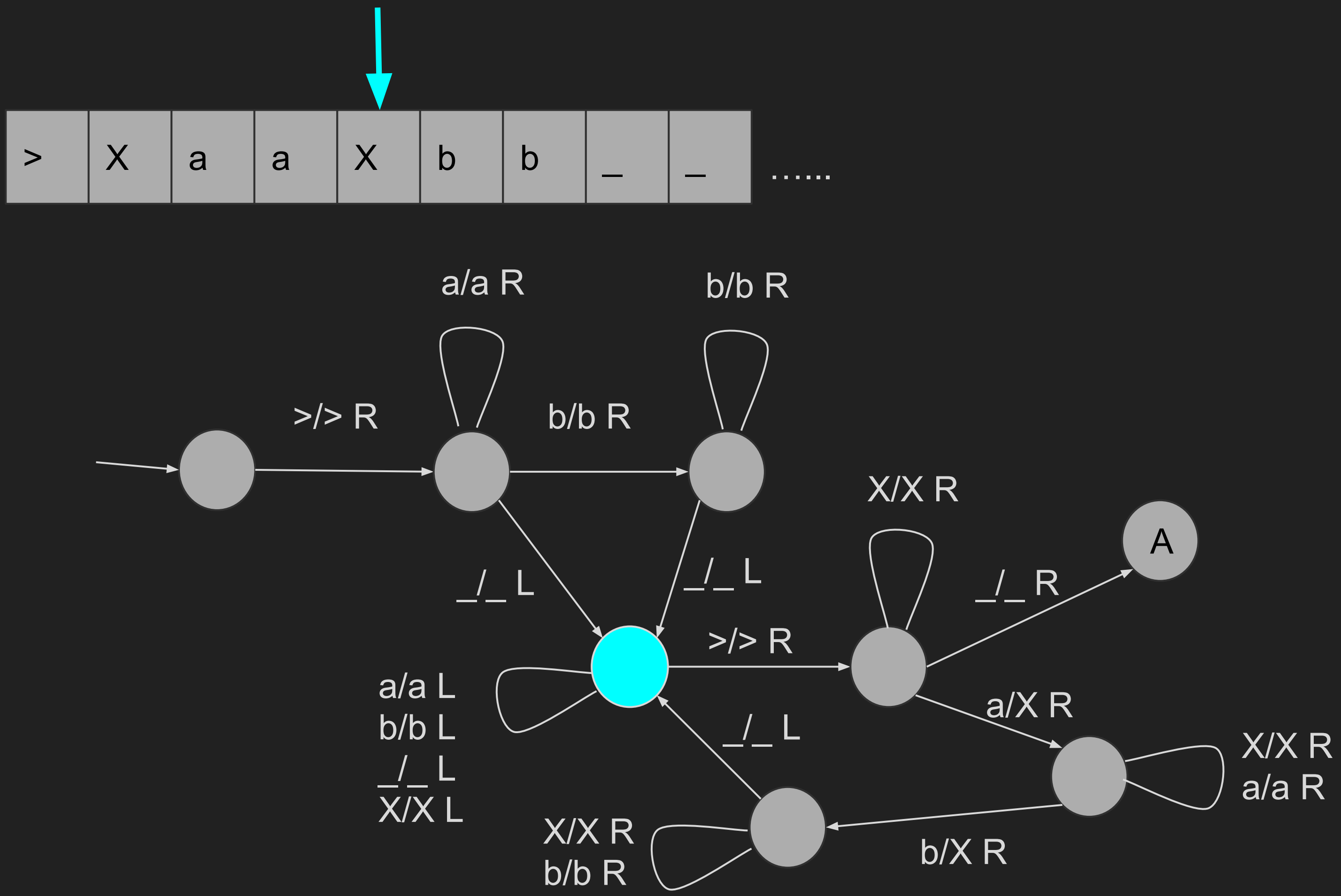
> X a a X b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



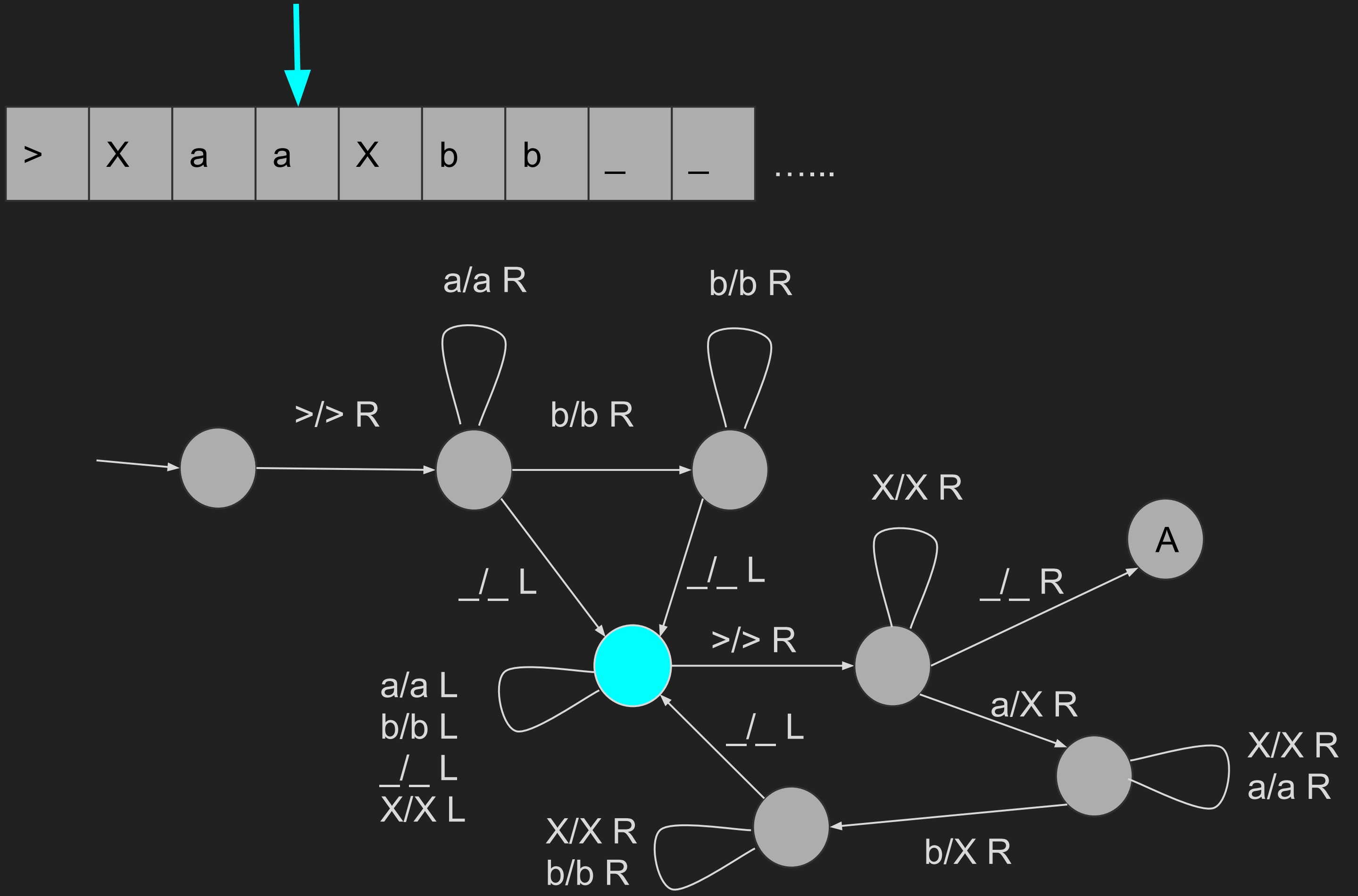
> X a a X b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



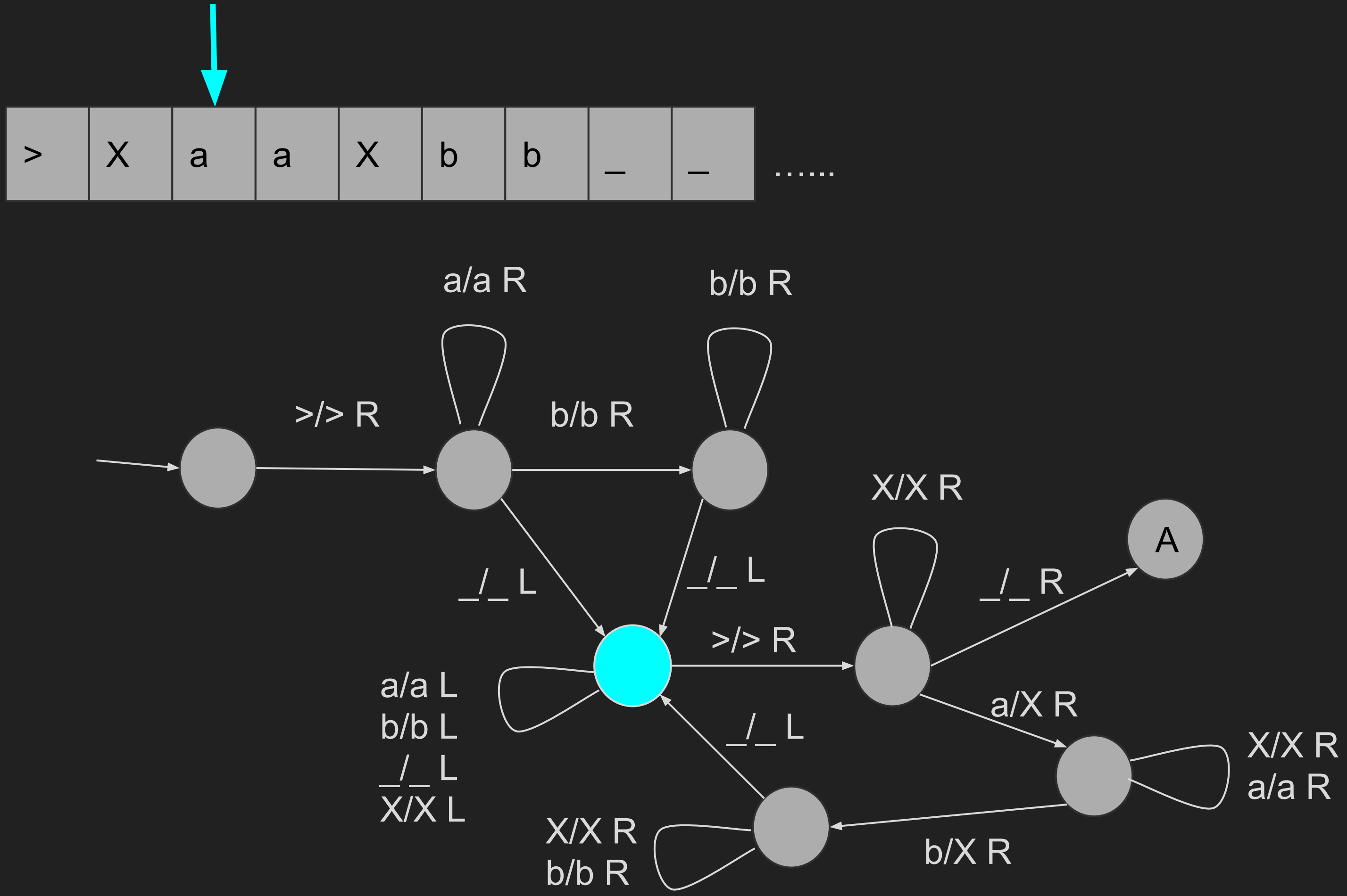
>
X
a
a
X
b
b
_
_
......
a/a R
b/b R
>/> R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



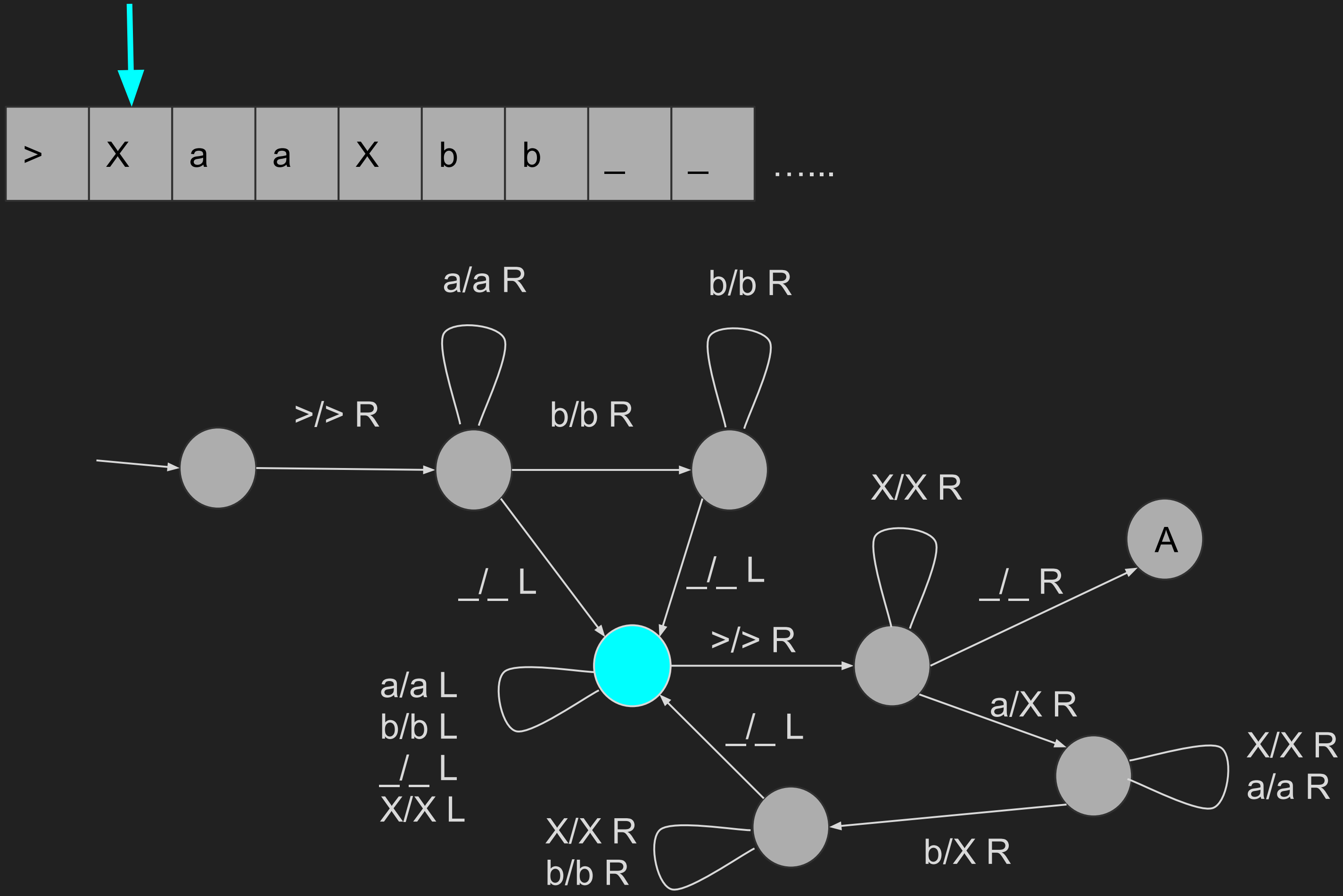
> X a a X b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



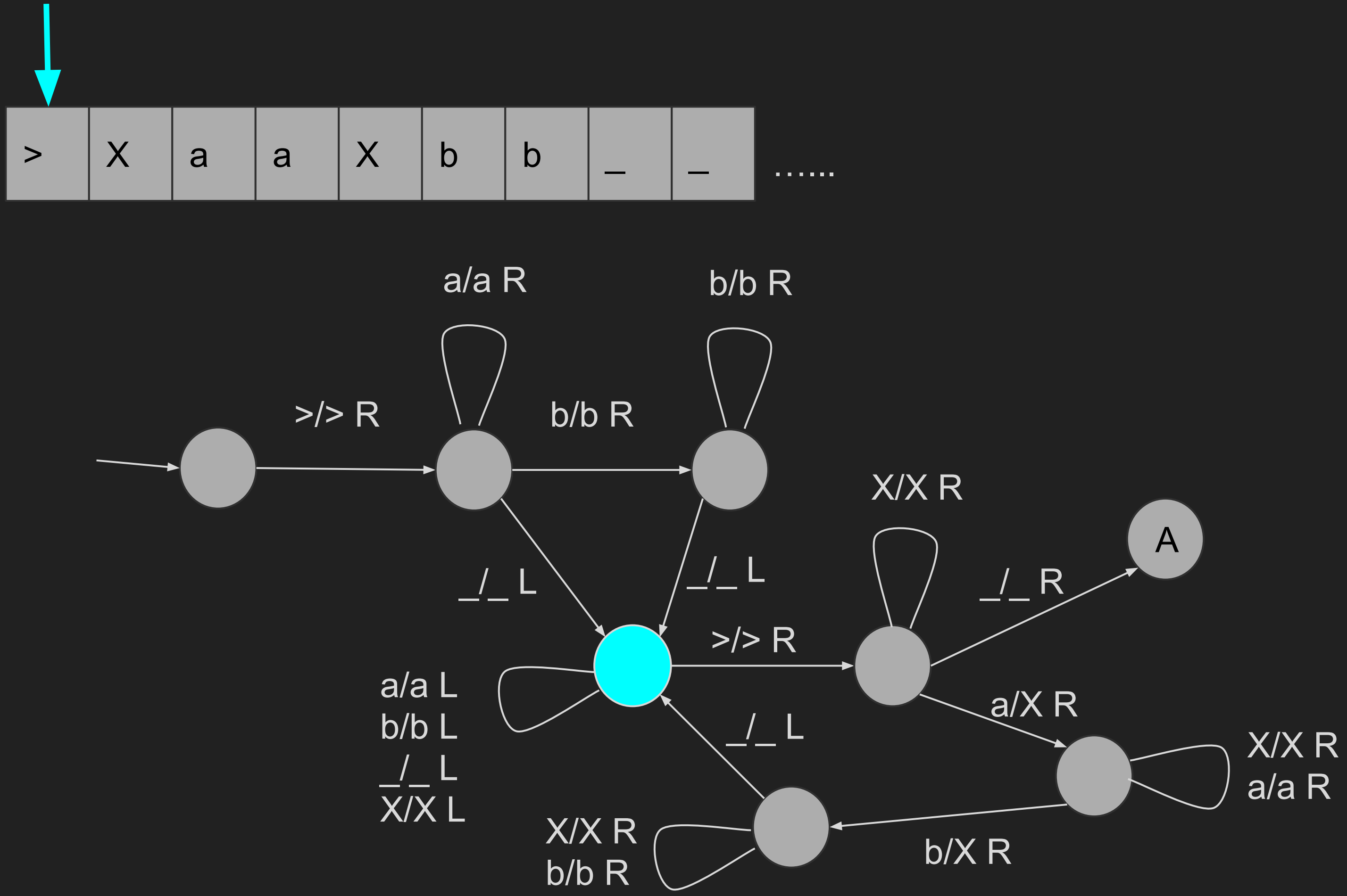
> X a a X b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



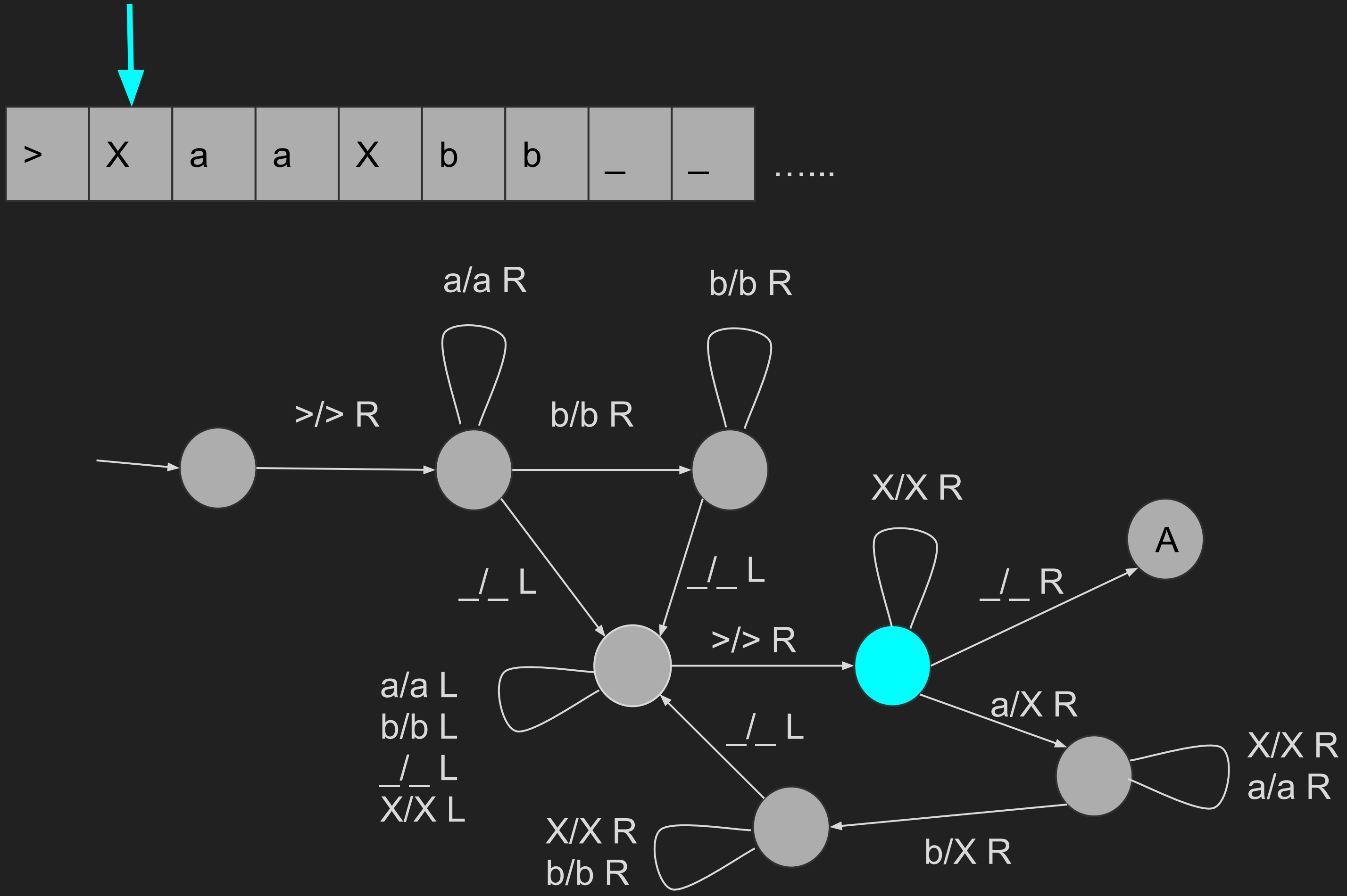
> X a a X b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

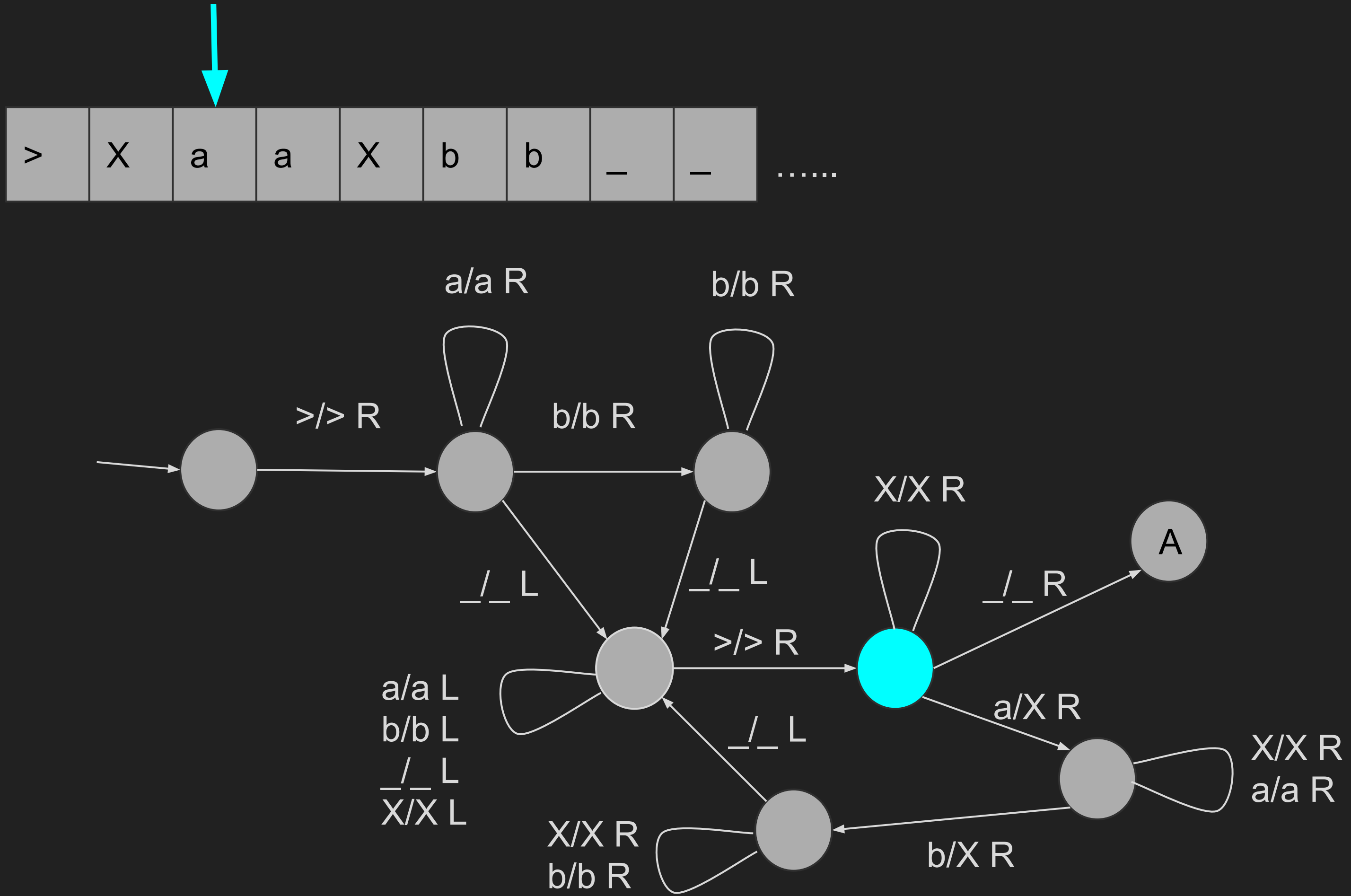


> X a a X b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

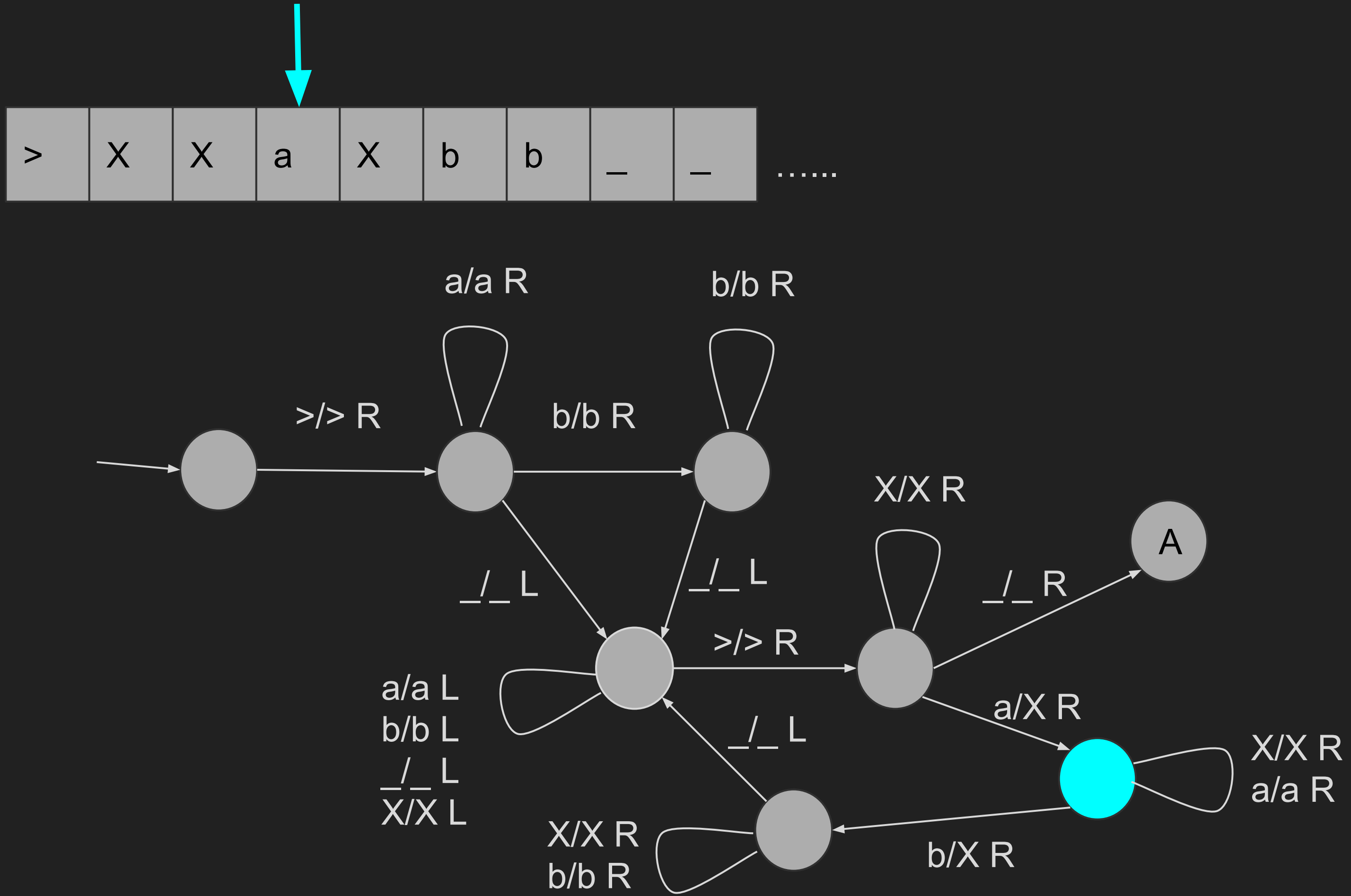


>
X
a
a
X
b
b
_
_
……
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

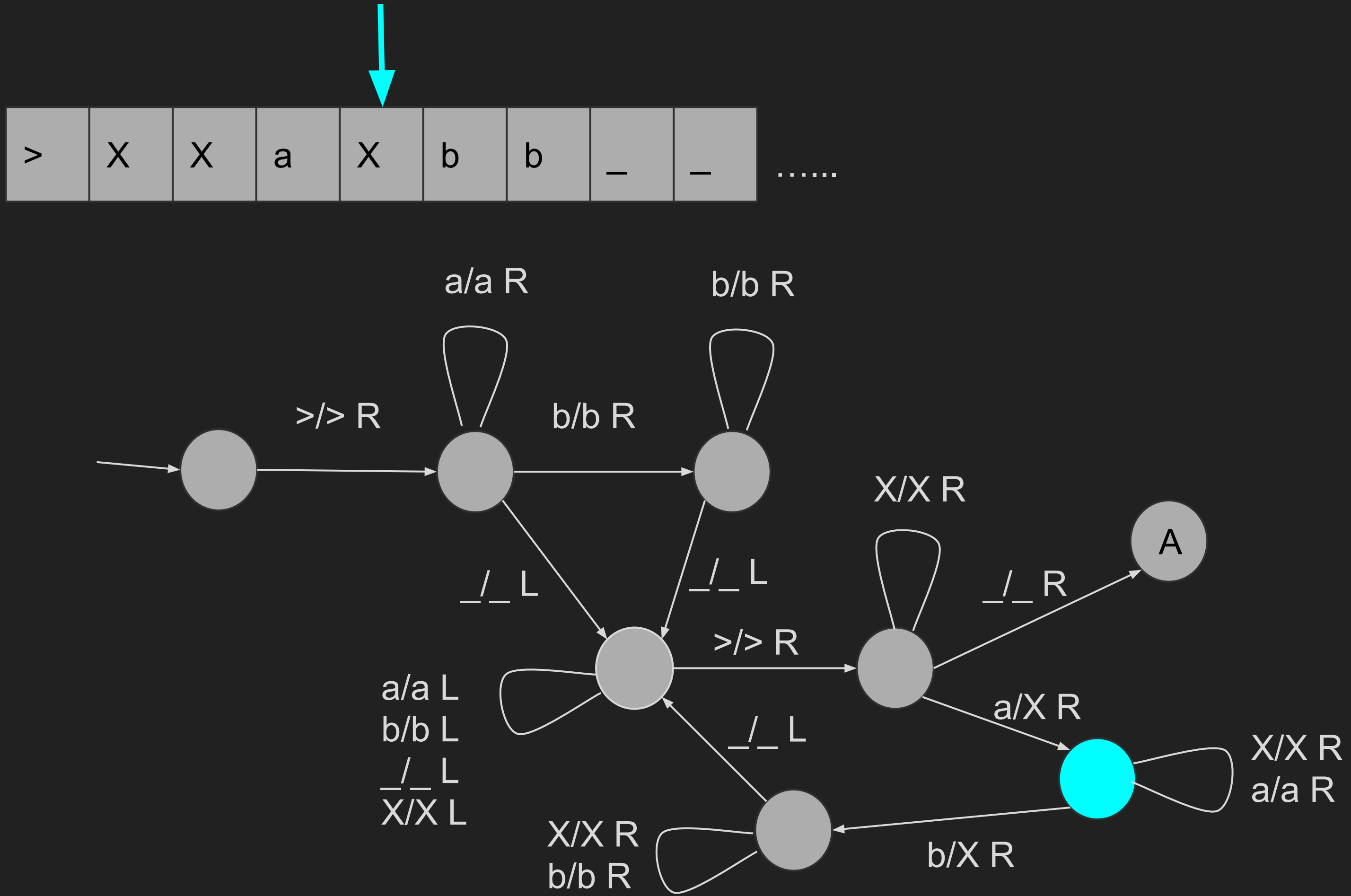

>
X
a
a
X
b
b
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



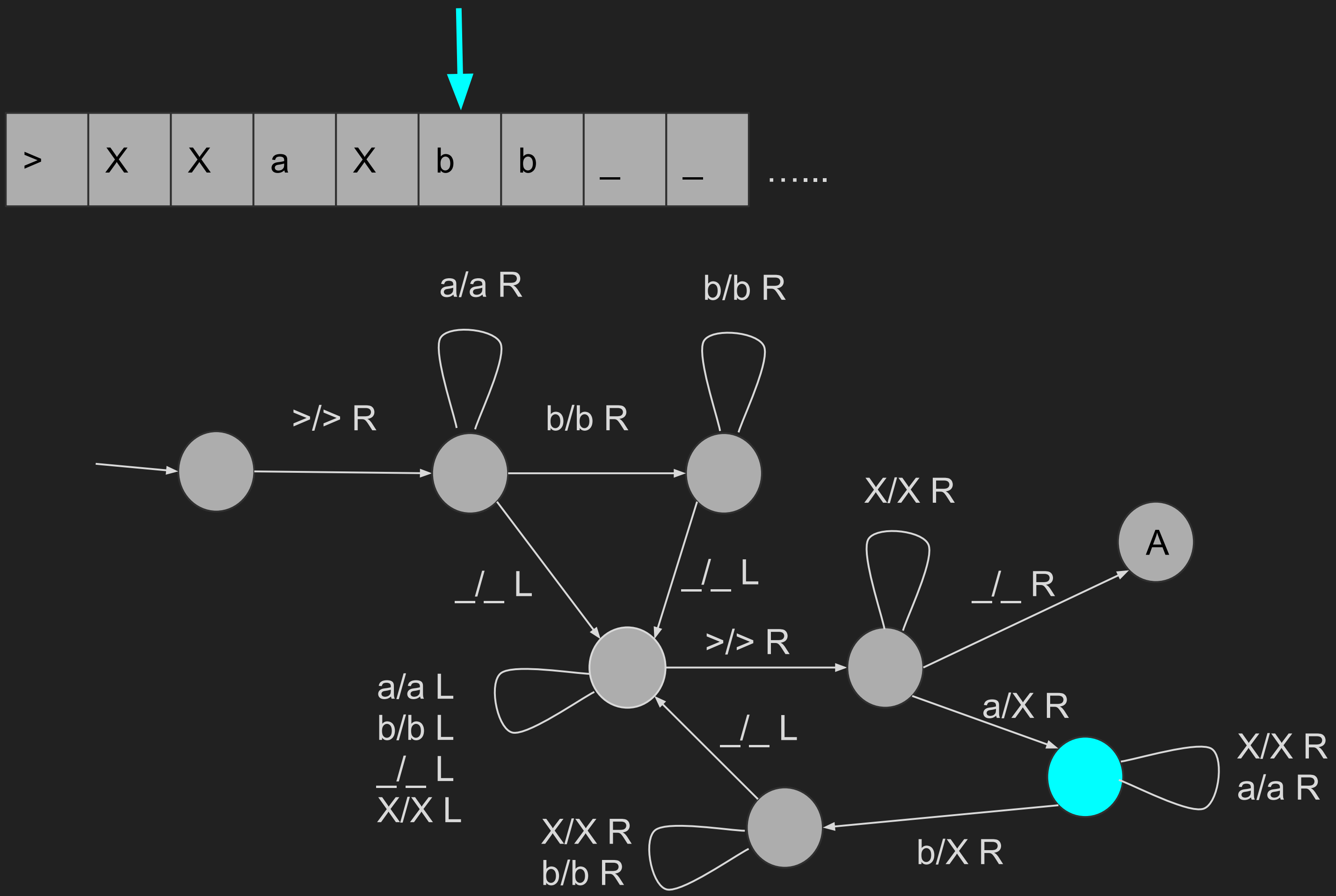
> X a a X b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



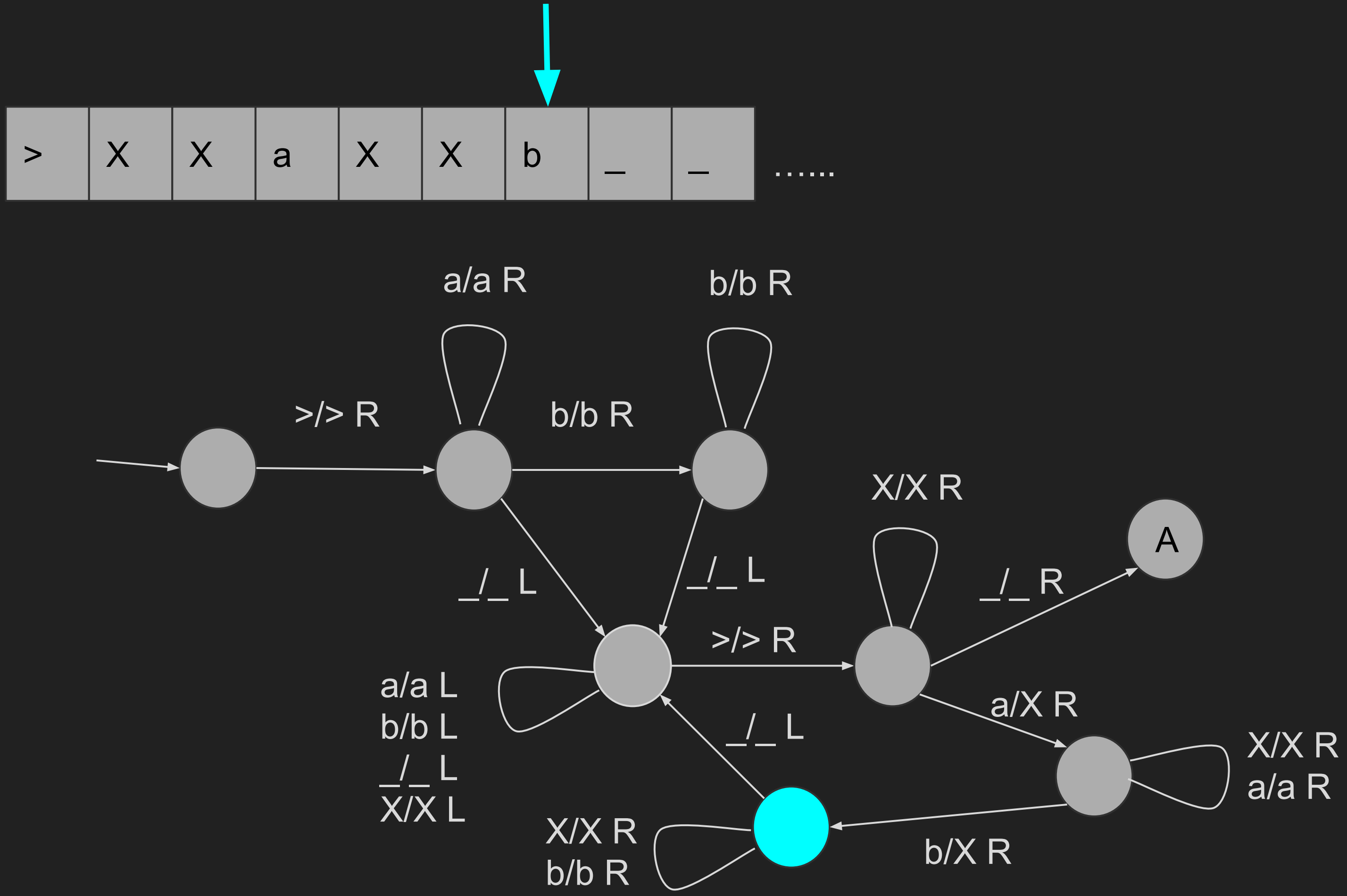
>
X
X
a
X
b
b
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
X/X R
a/a R
_/_ L
X/X R
b/b R
b/X R



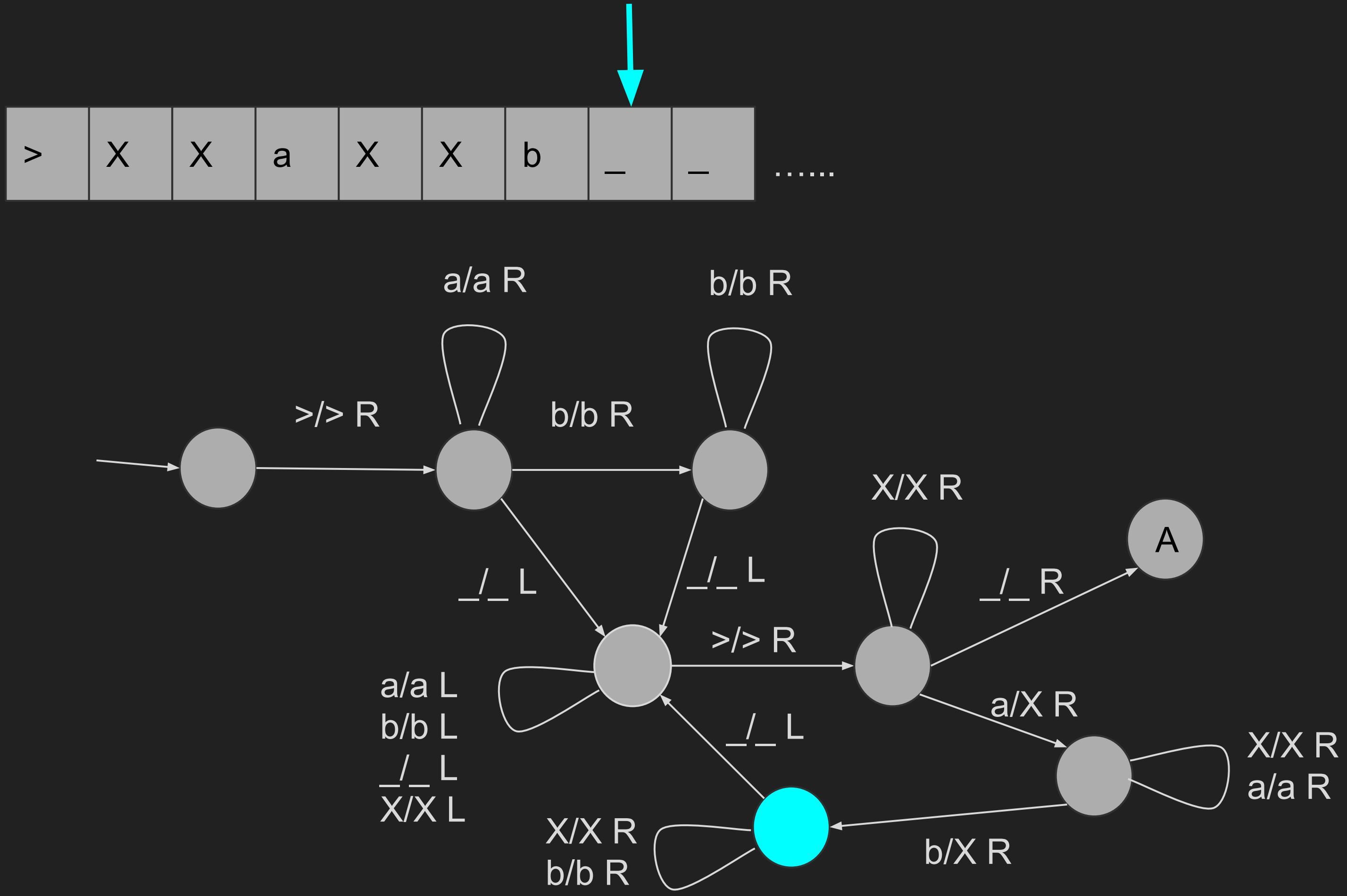
> X X a X b b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
X/X R
a/a R
_/_ L
X/X R
b/b R
b/X R



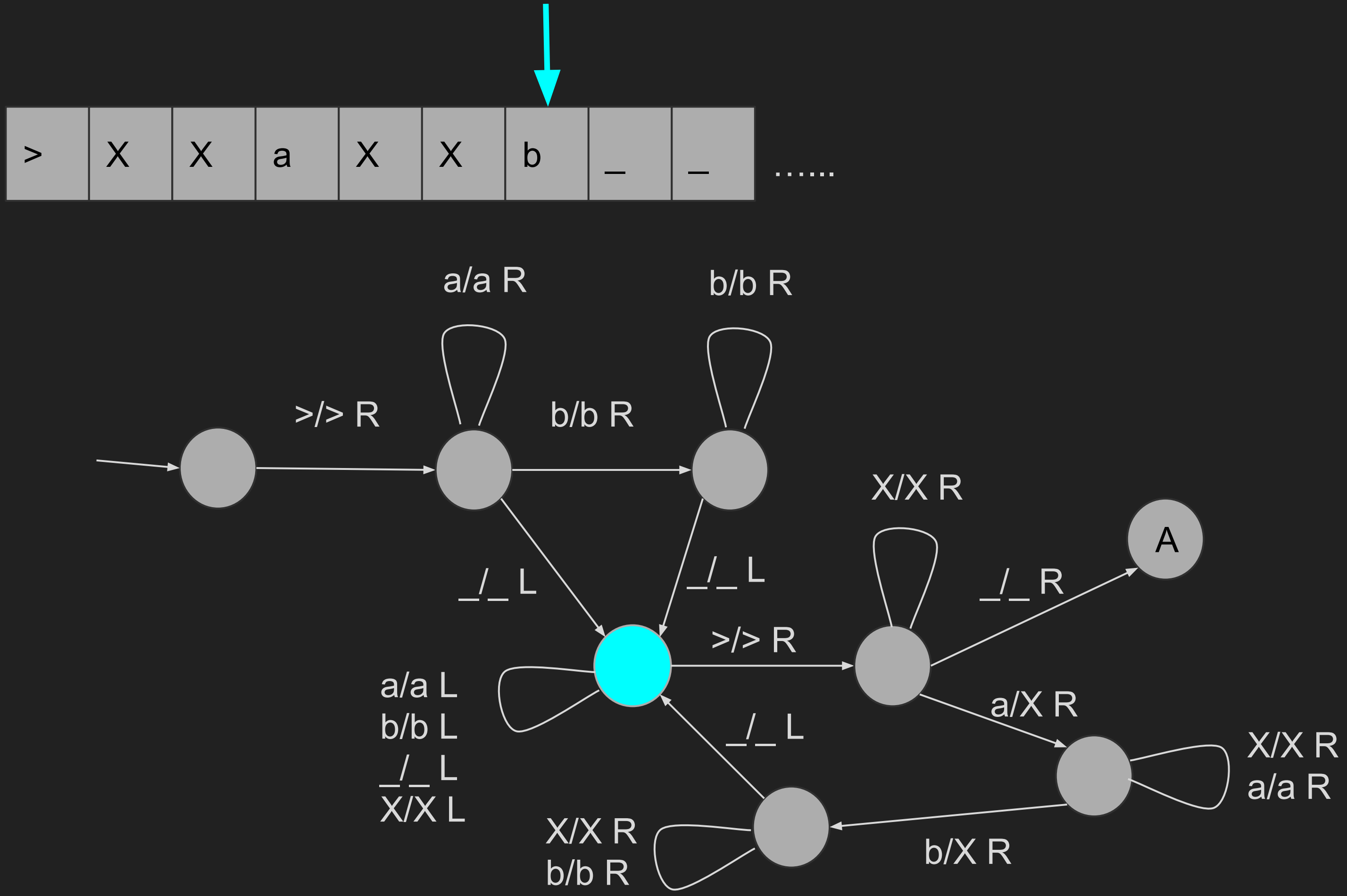
> X X a X b b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



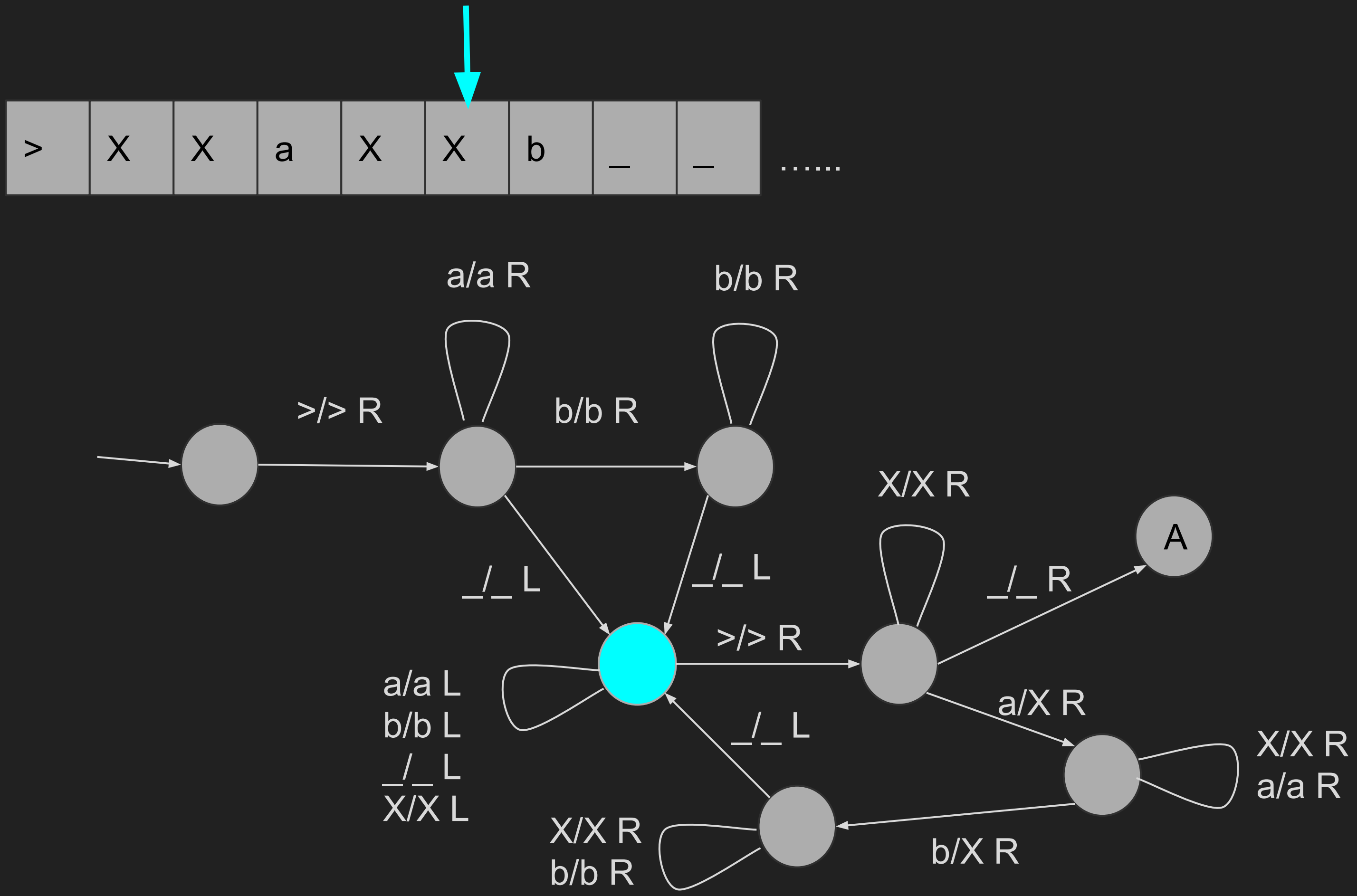
>
X
X
a
X
X
b
_
_
......
a/a R
b/b R
>/> R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



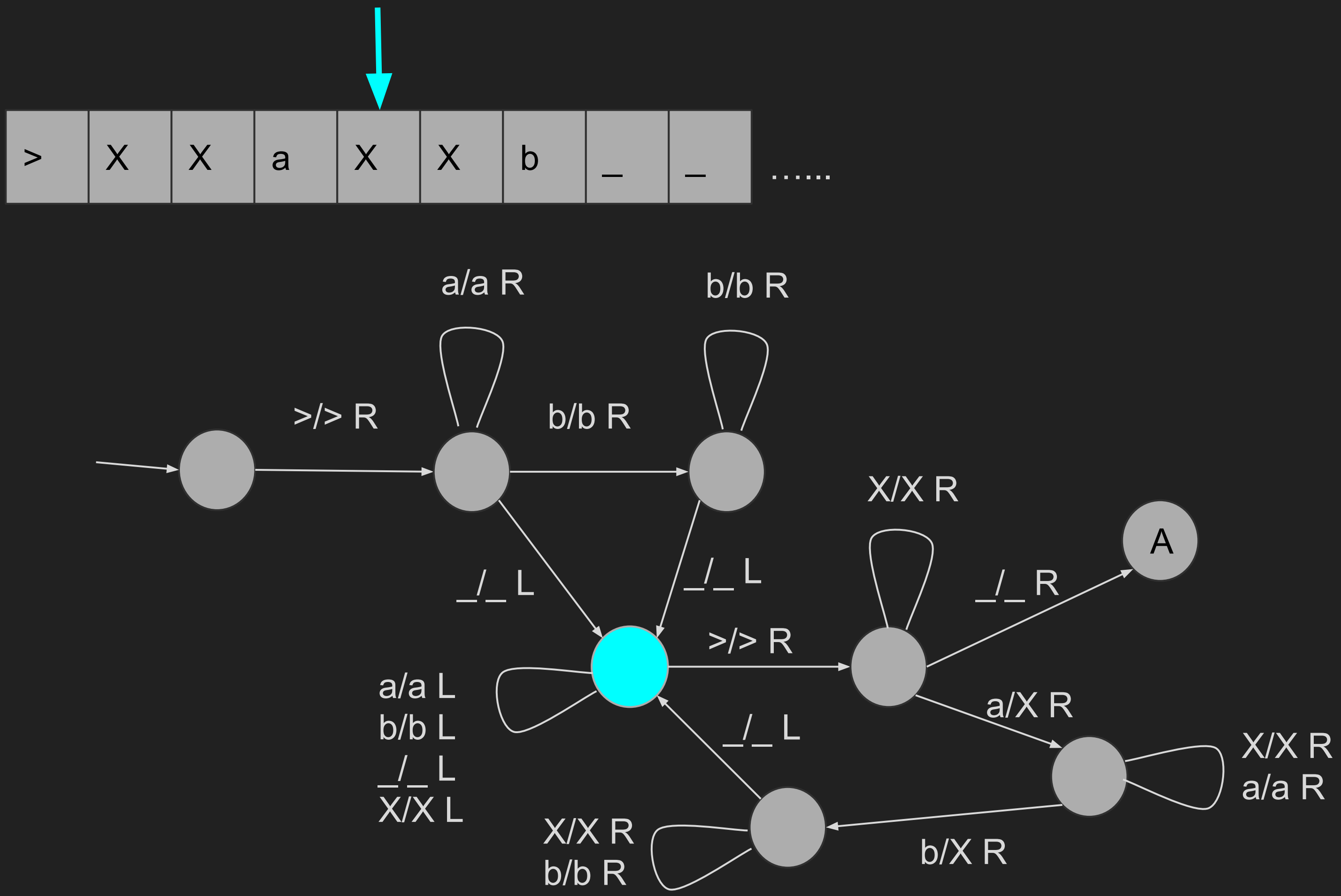
>
X
X
a
X
X
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



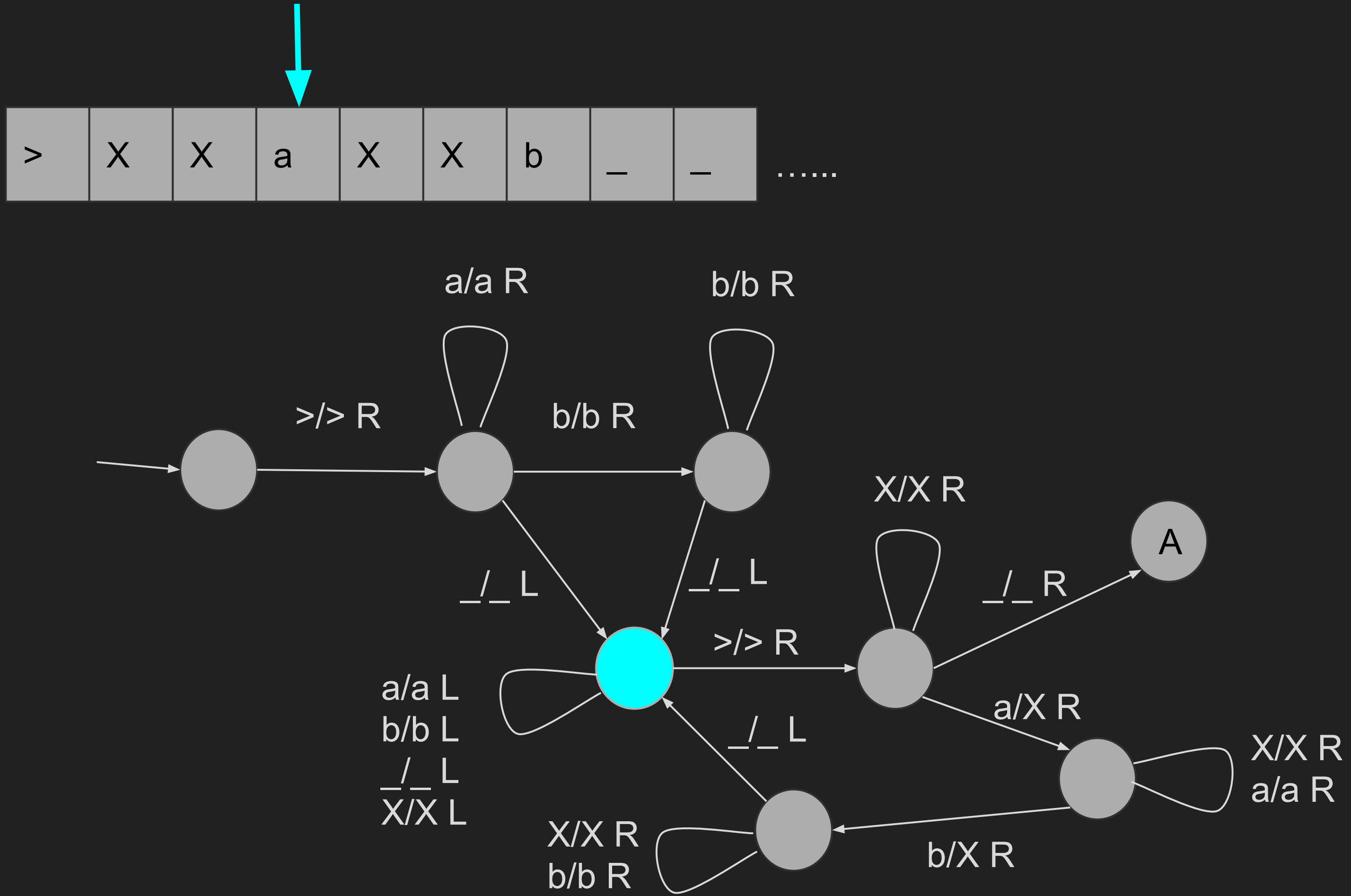
> X X a X X b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



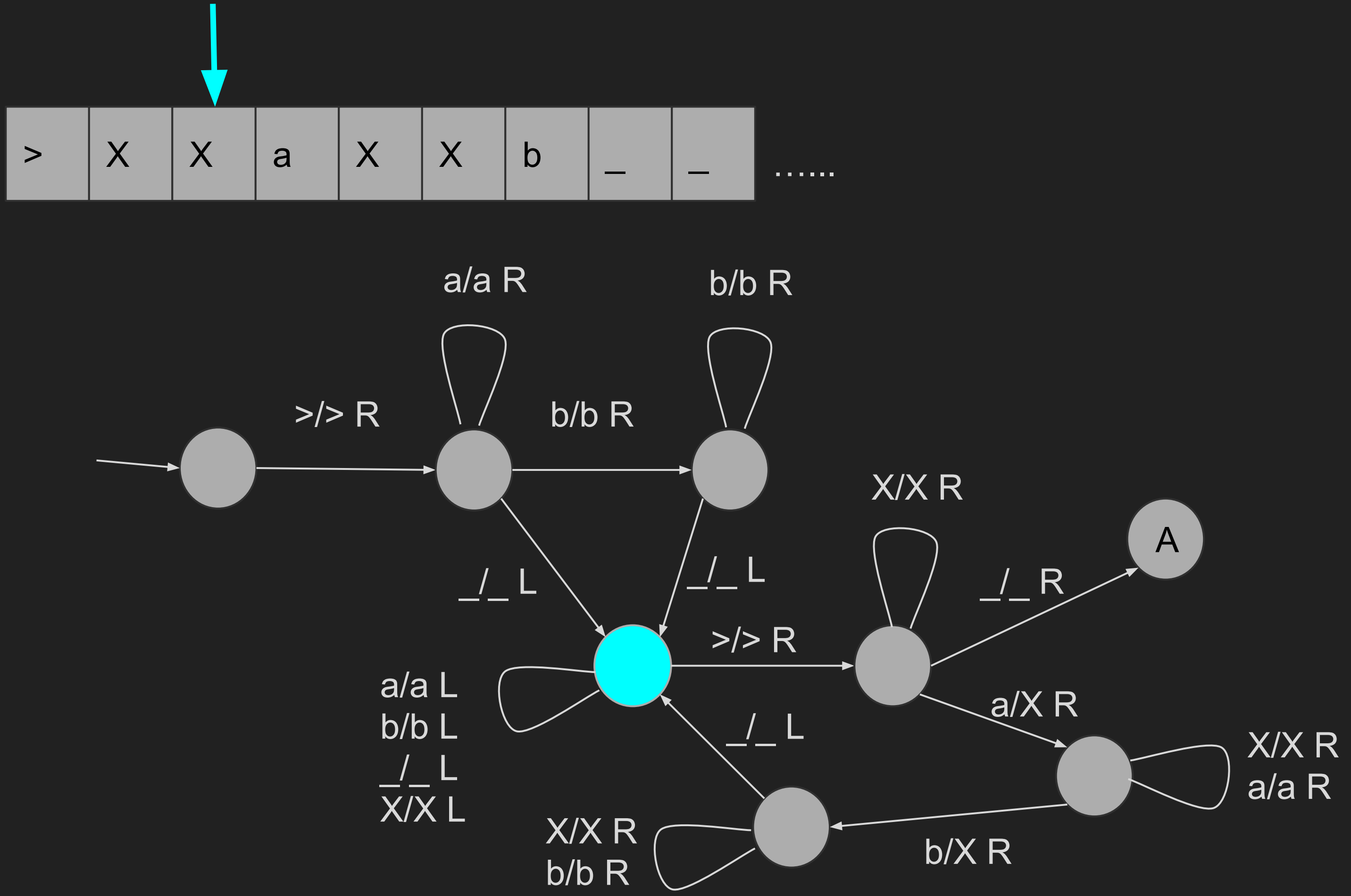
>
X
X
a
X
X
b
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
X/X R
a/a R
_/_ L
X/X R
b/b R
b/X R

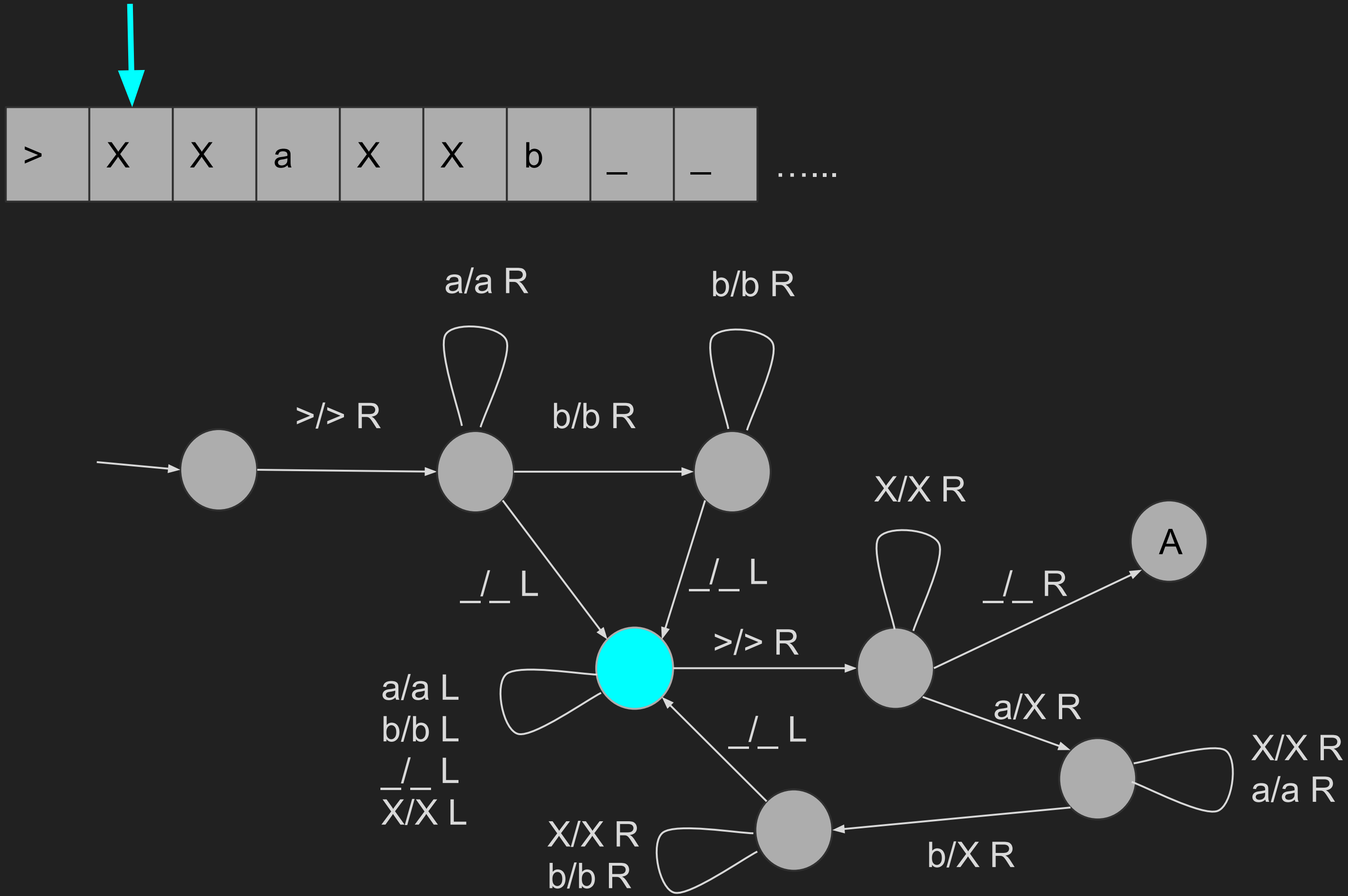


> X X a X X b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

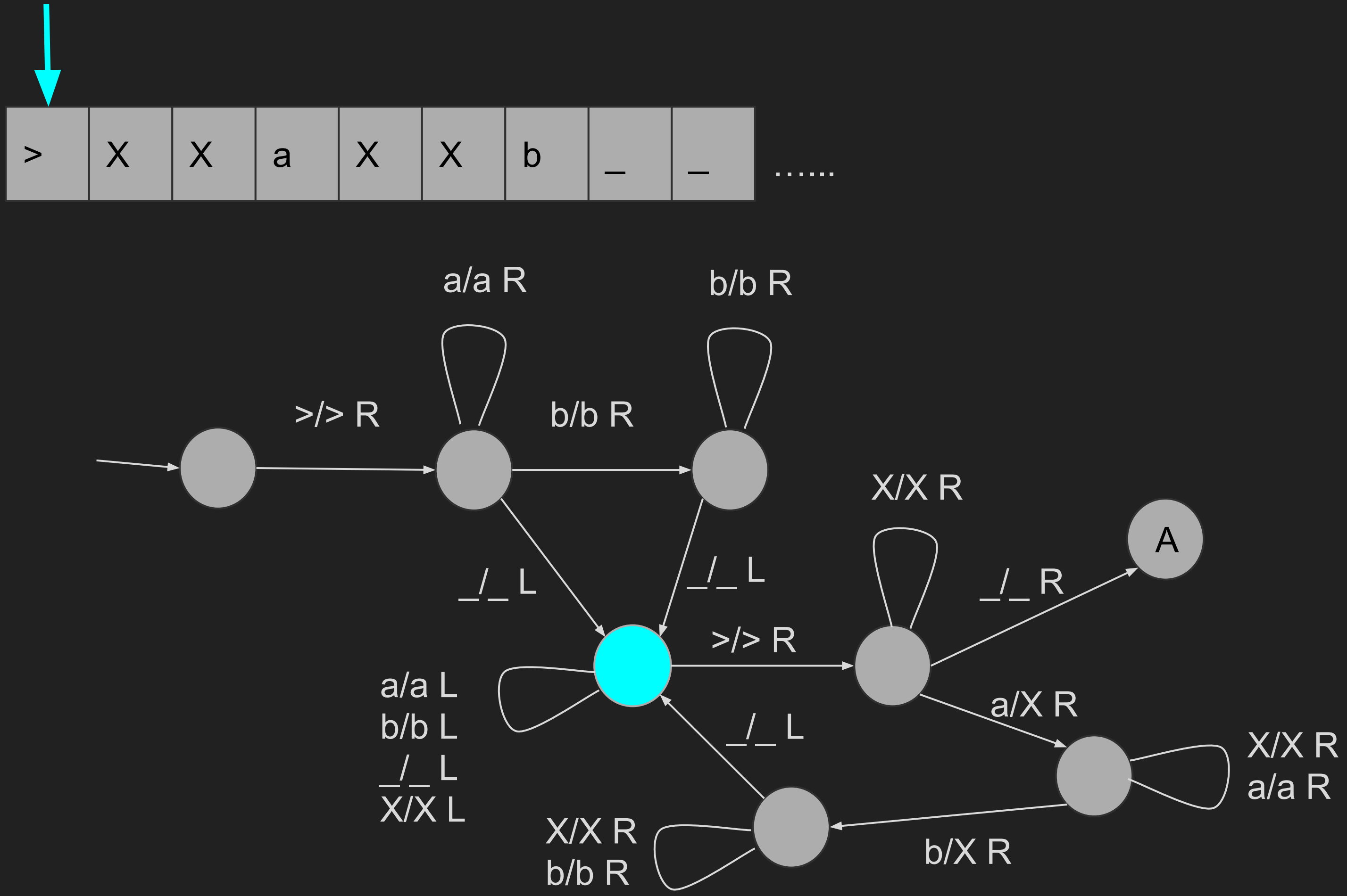


>
X
X
a
X
X
b
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

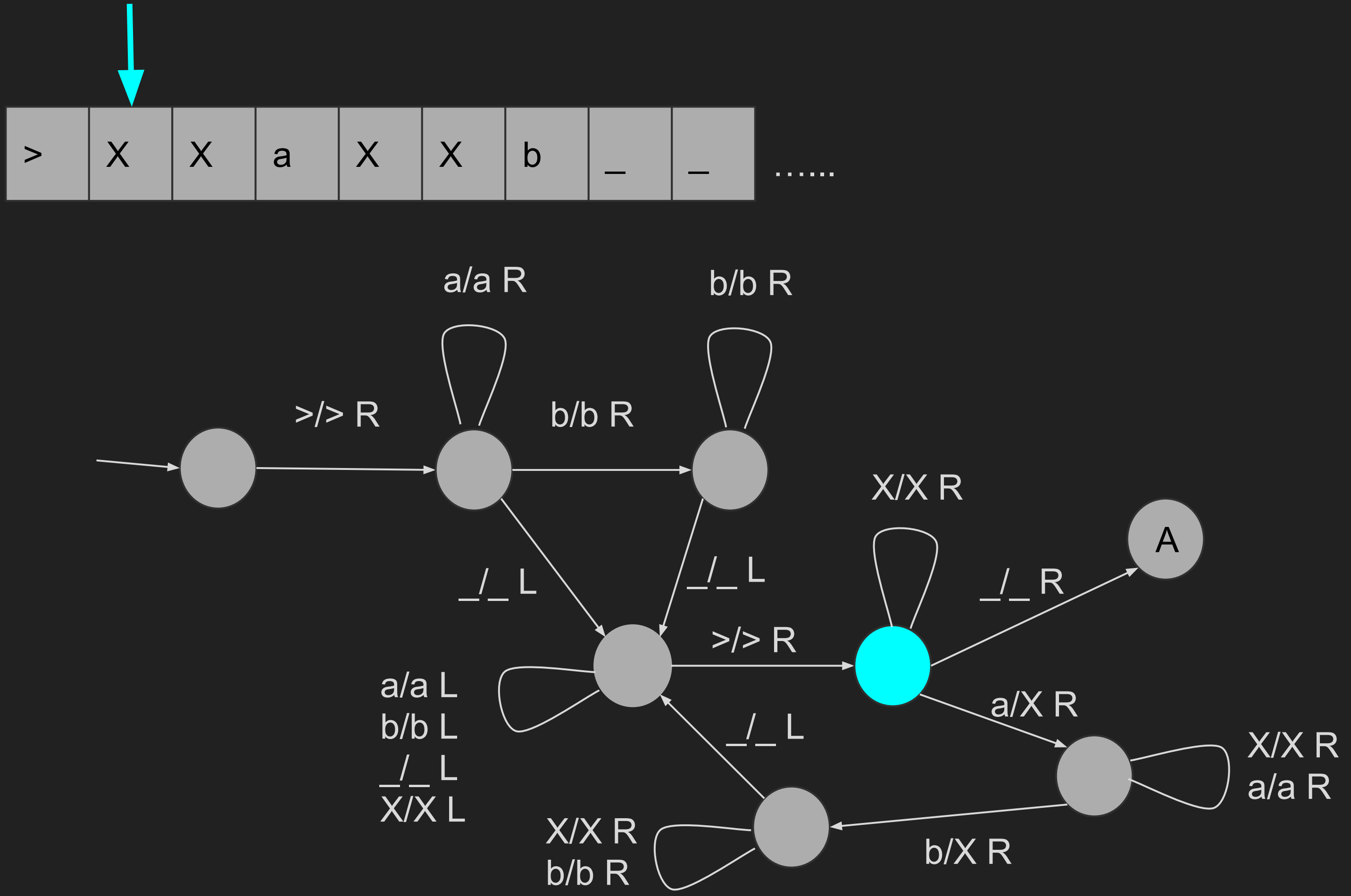

>
X
X
a
X
X
b
_
_
……
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



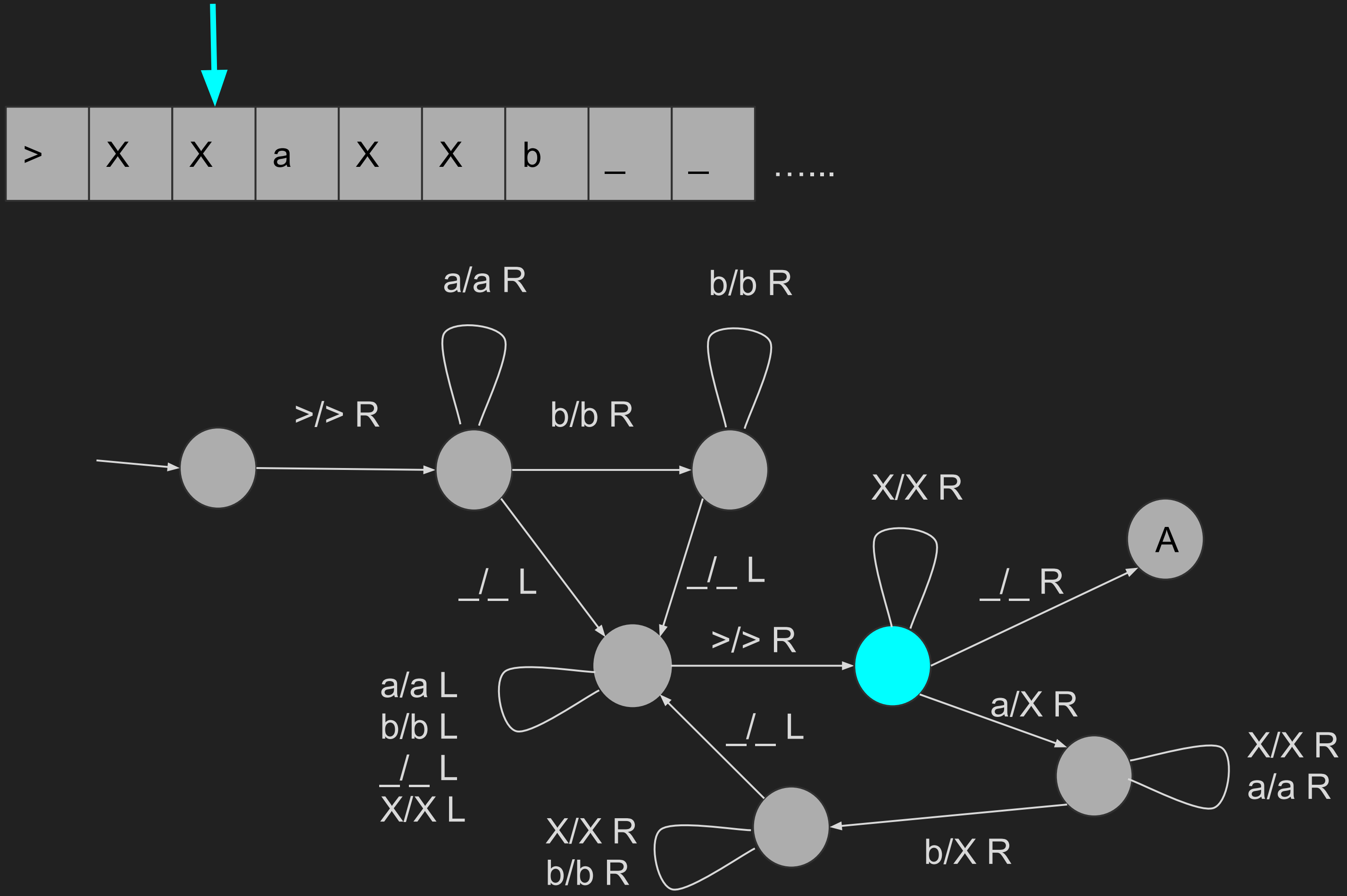
>
X
X
a
X
X
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



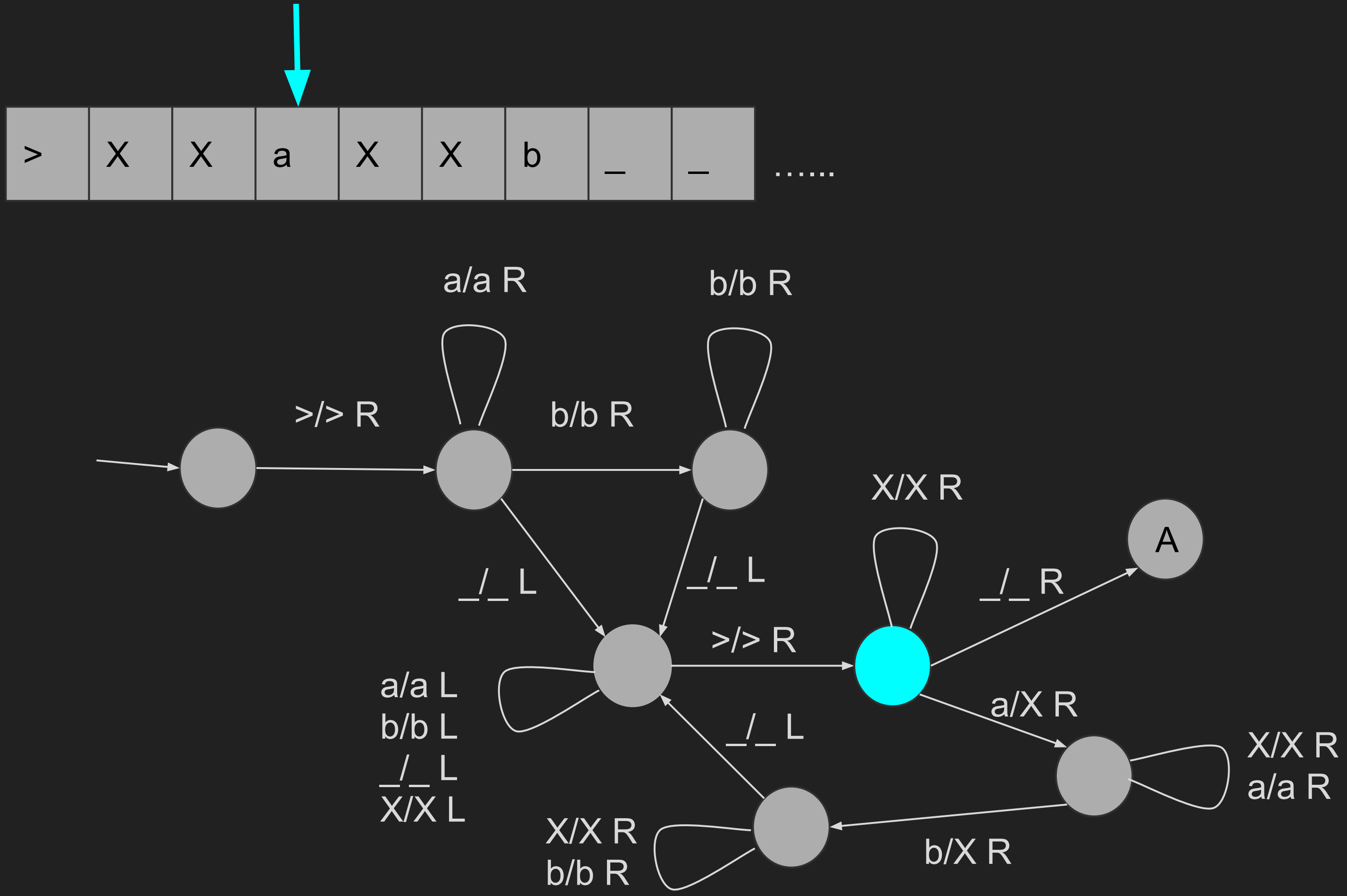
> X X a X X b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

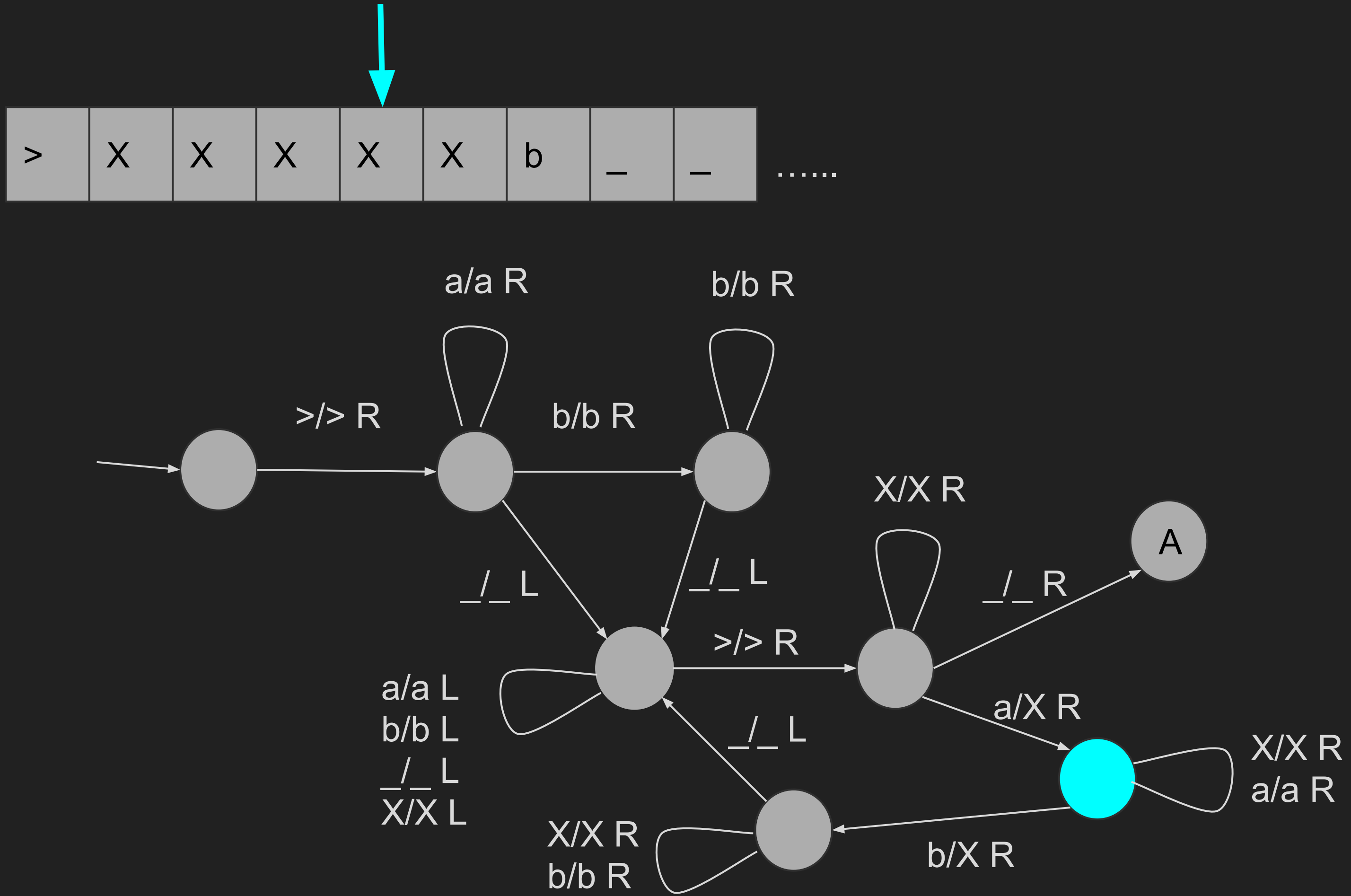


> X X a X X b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

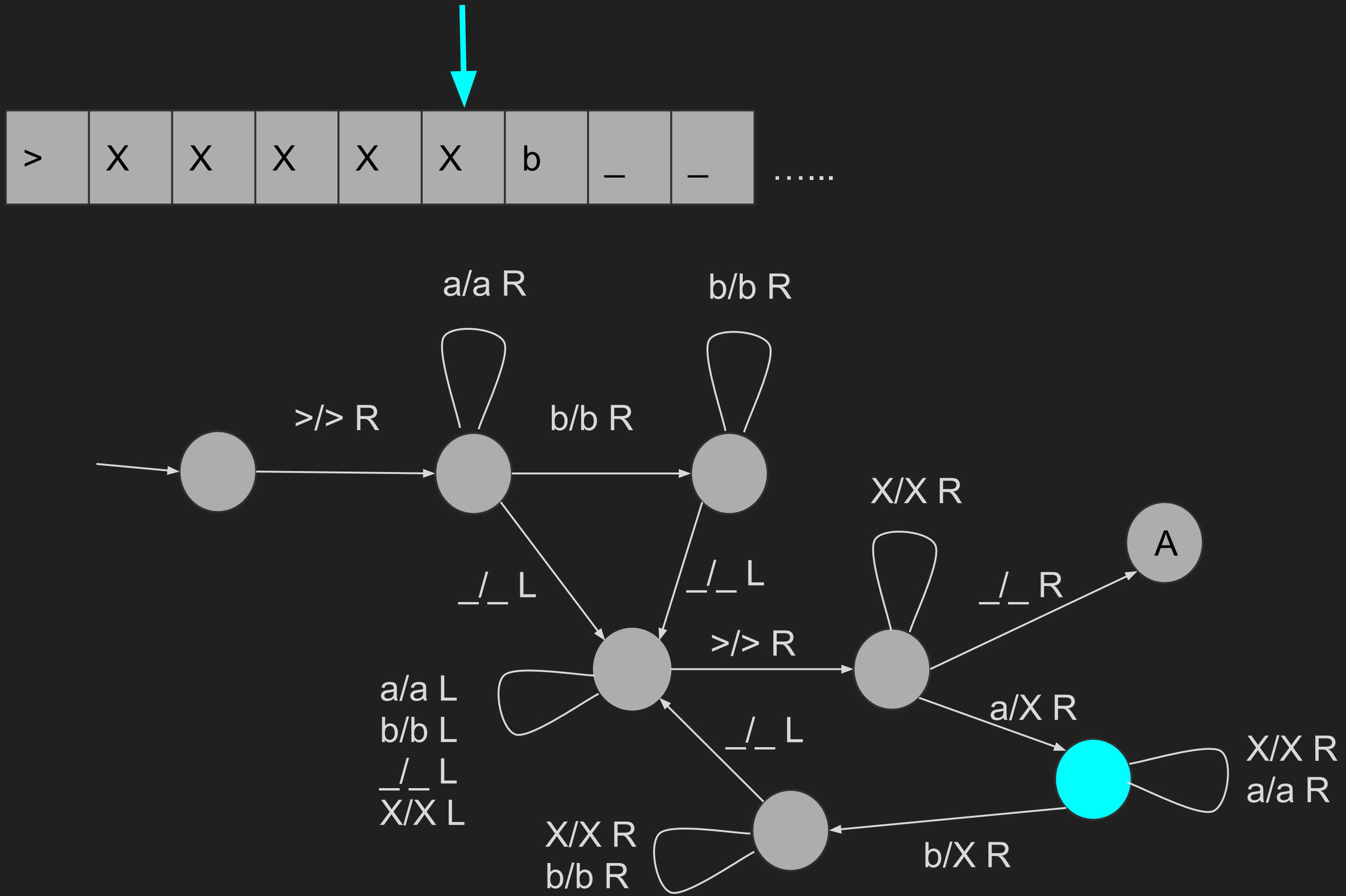


> X X a X X b _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

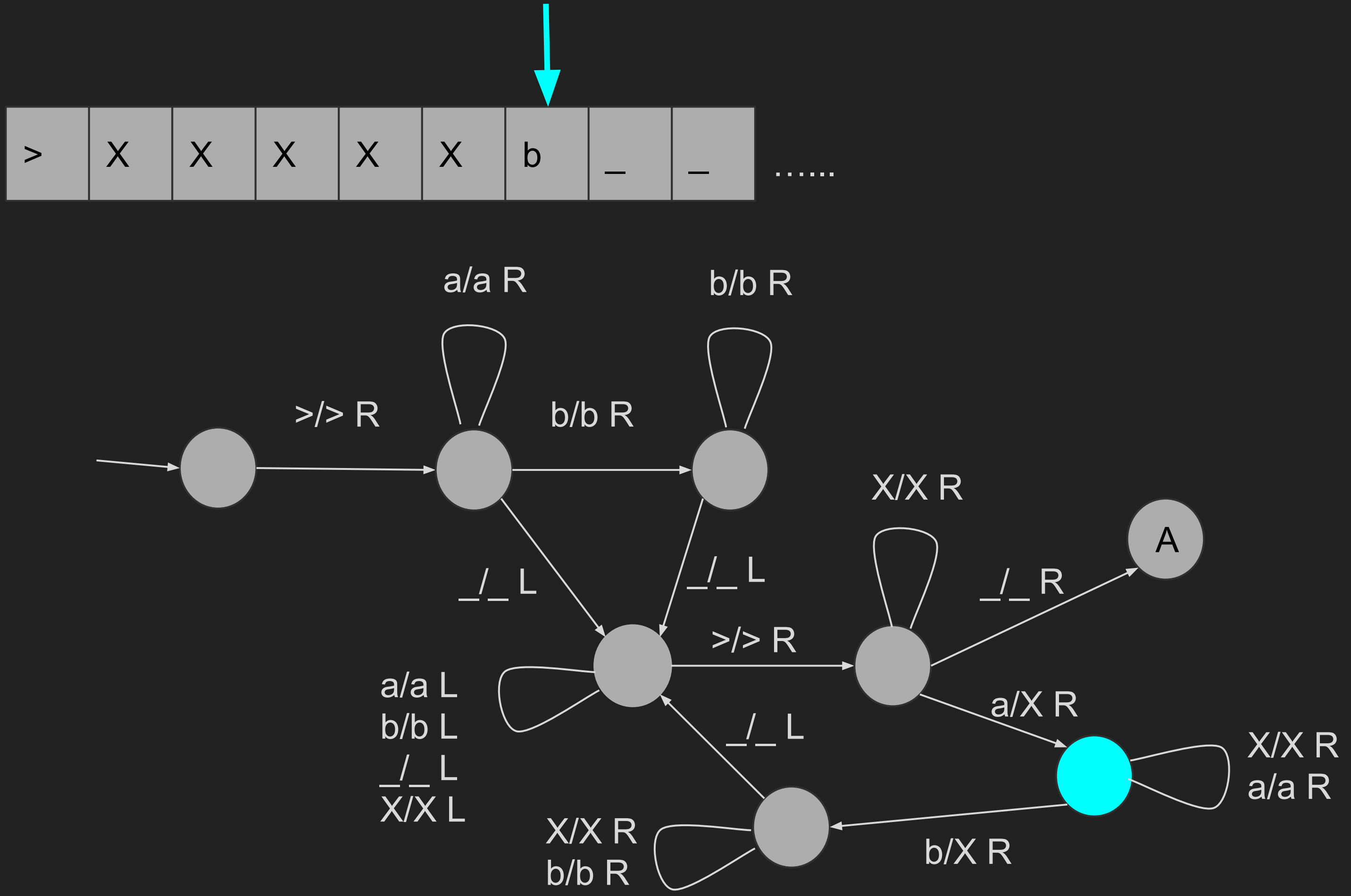

> X X a X X b _ _ ......
a/a R
b/b R
>/> R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



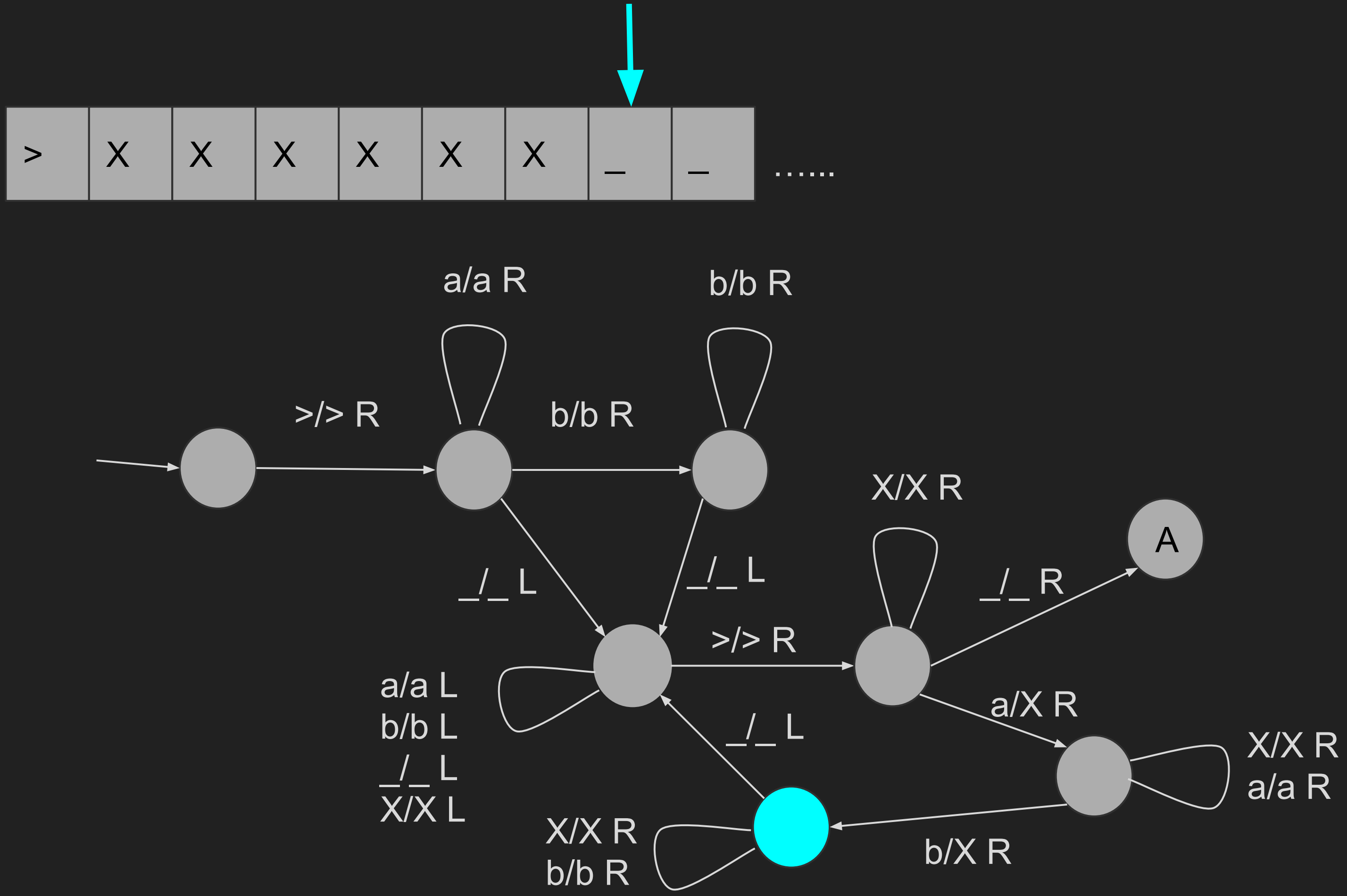
> X X X X X b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



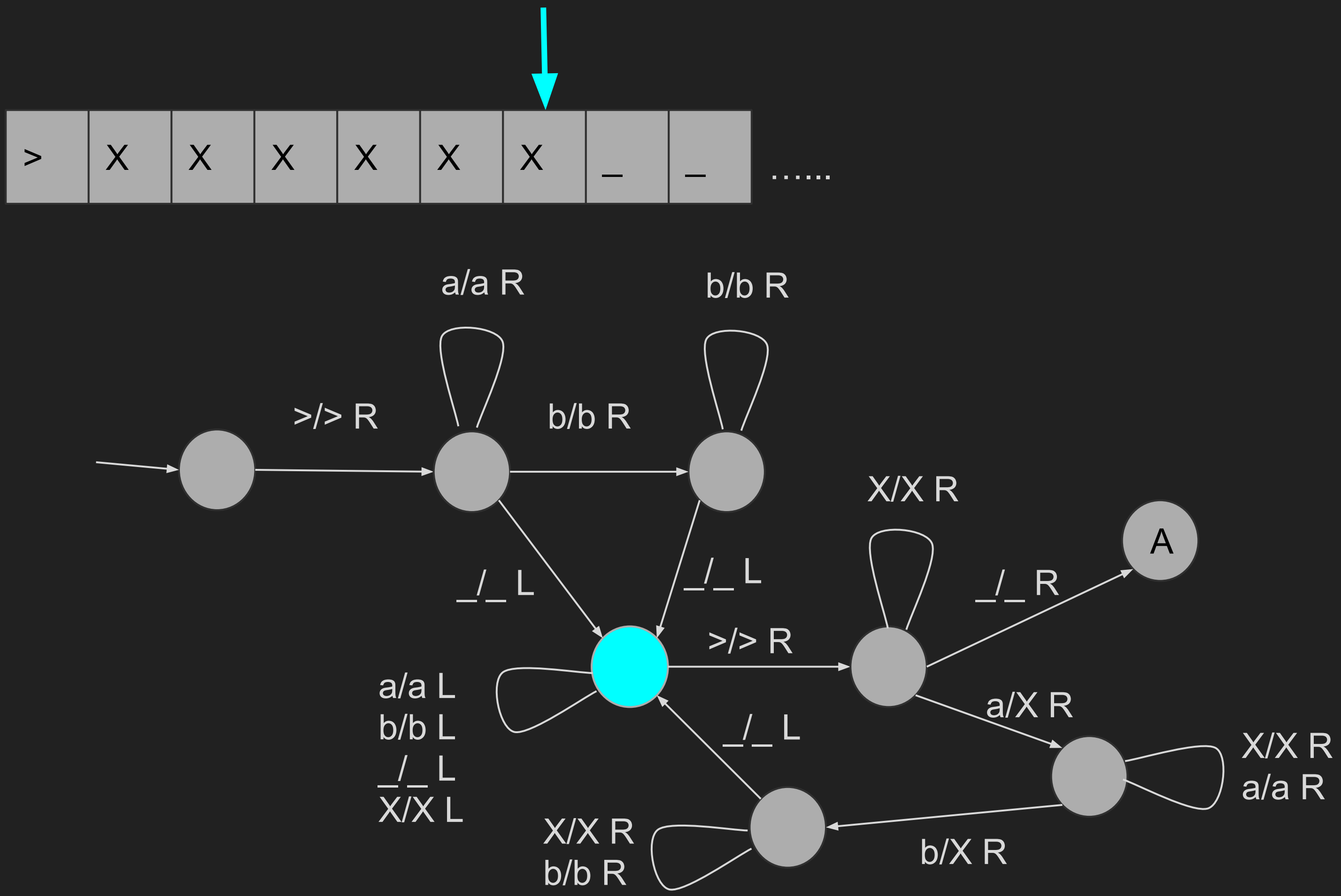
> X X X X X b _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



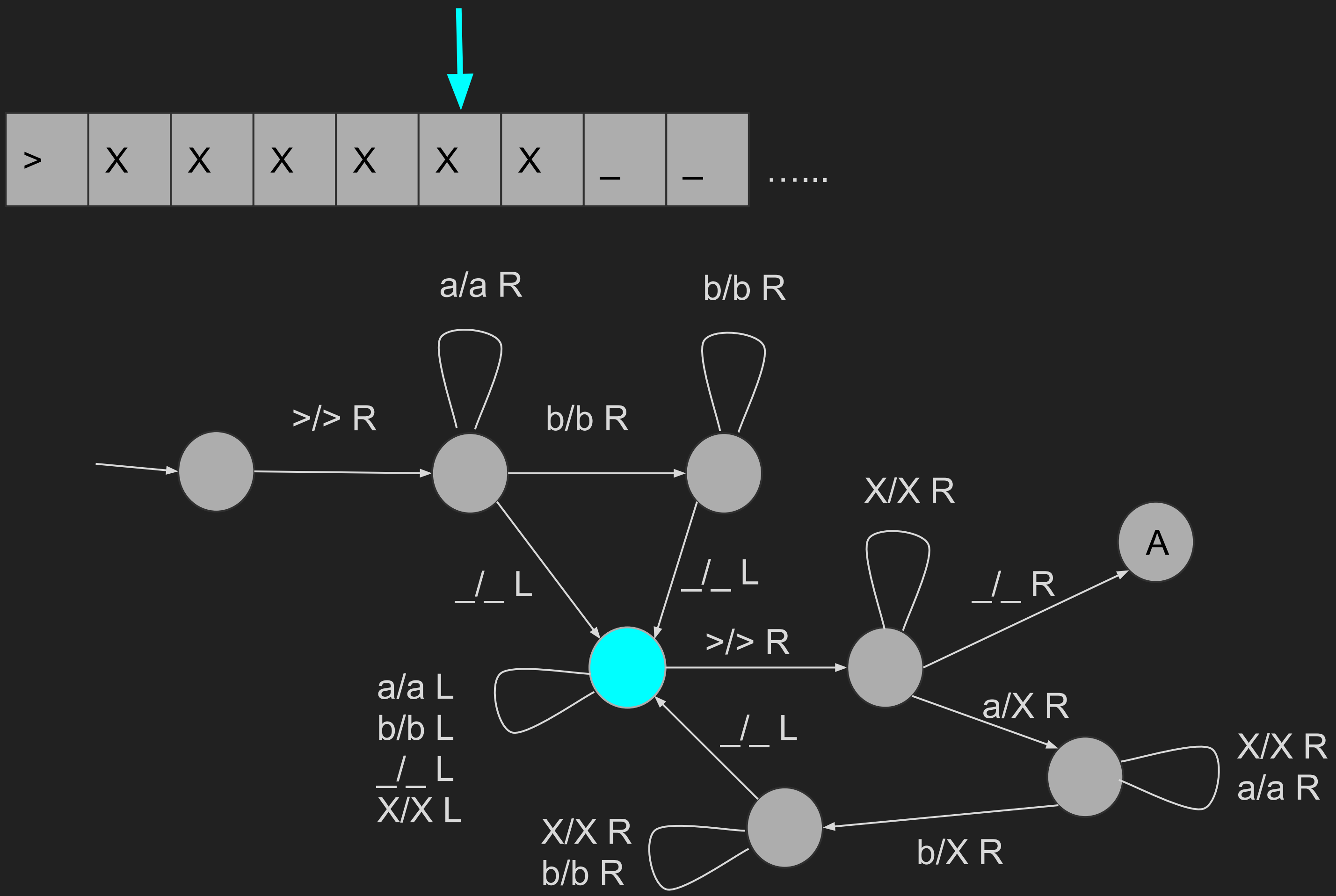
>
X
X
X
X
X
b
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



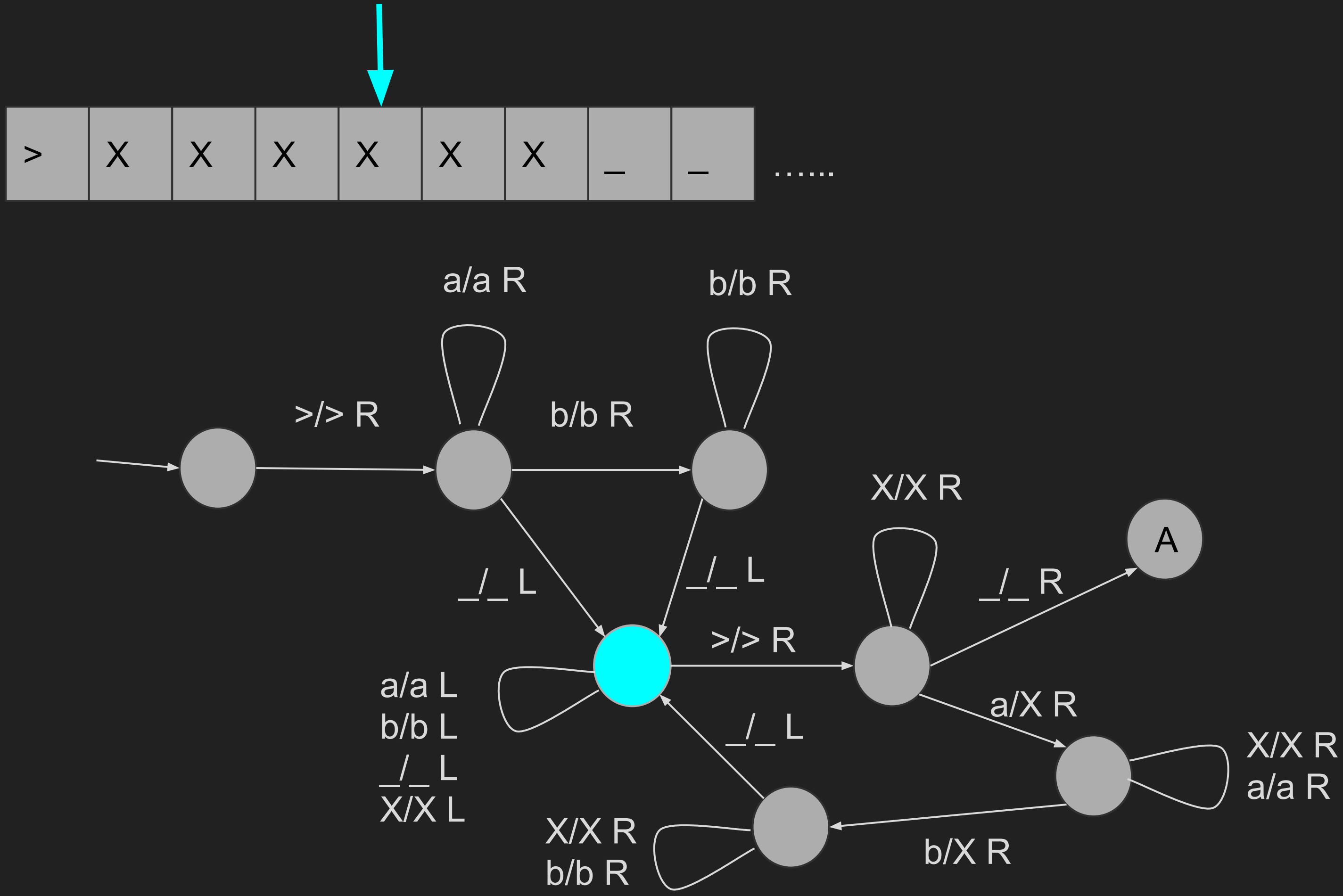
> X X X X X X _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



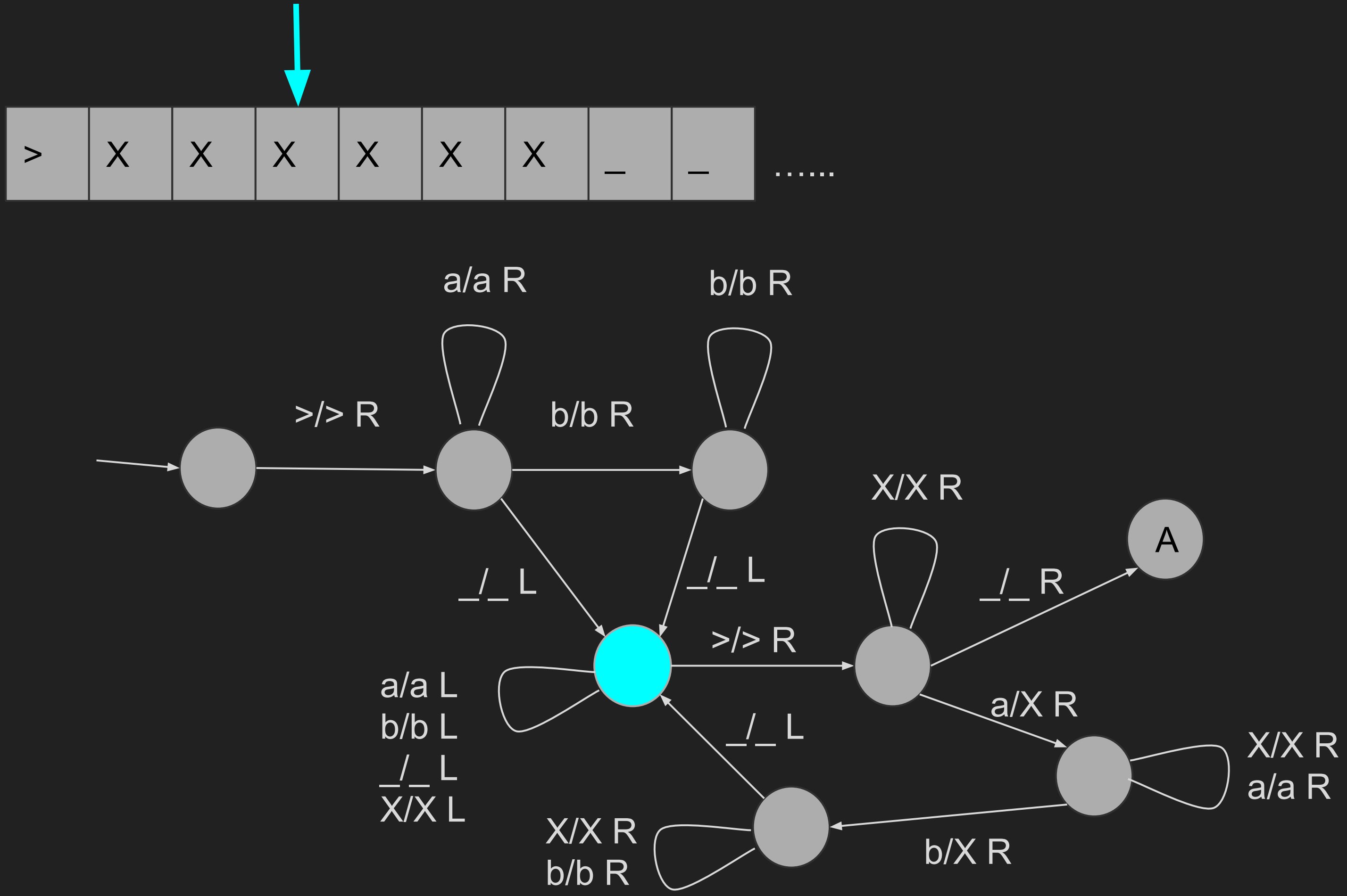
>
X
X
X
X
X
X
_
_
……
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



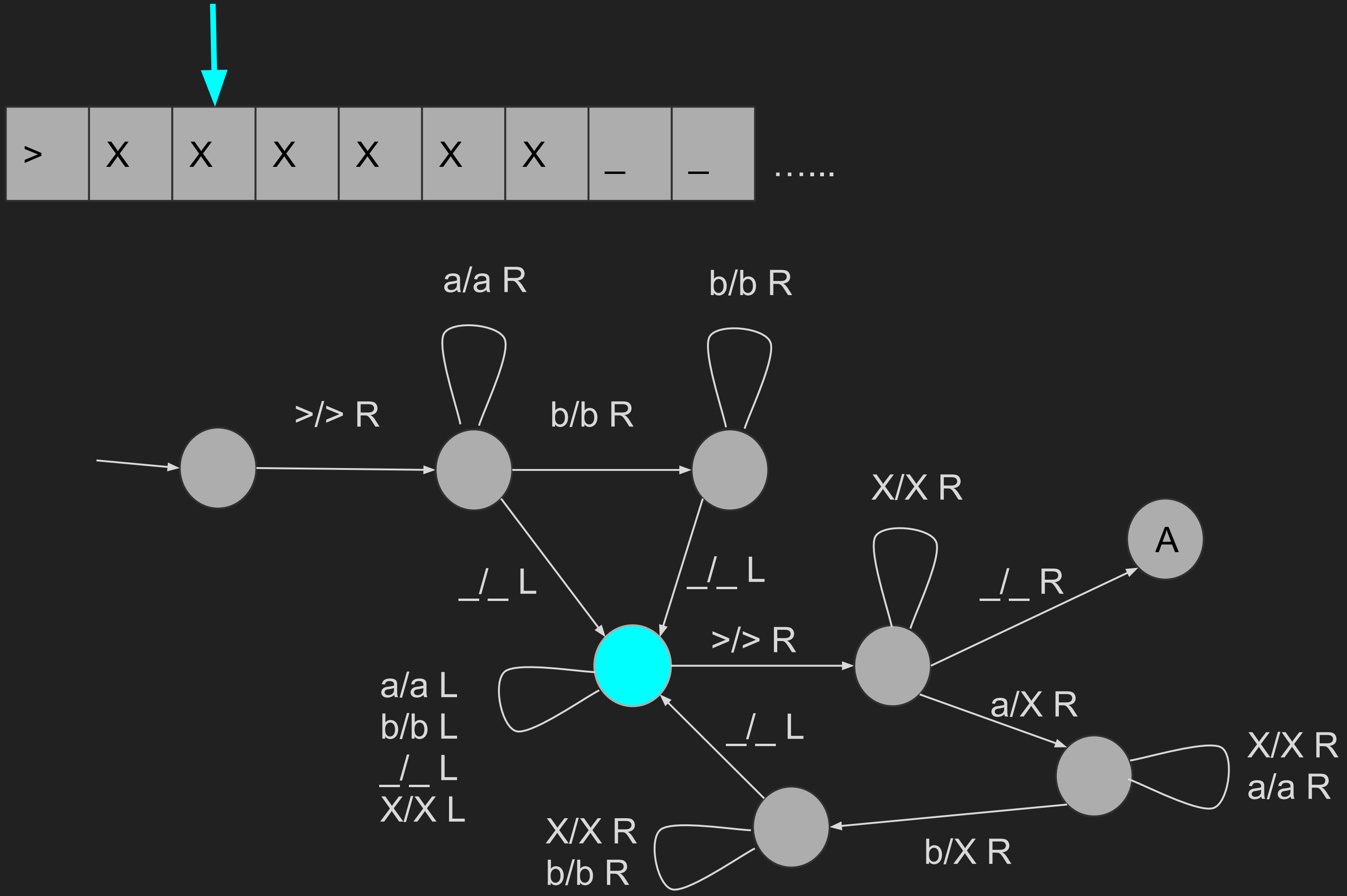
>
X
X
X
X
X
X
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



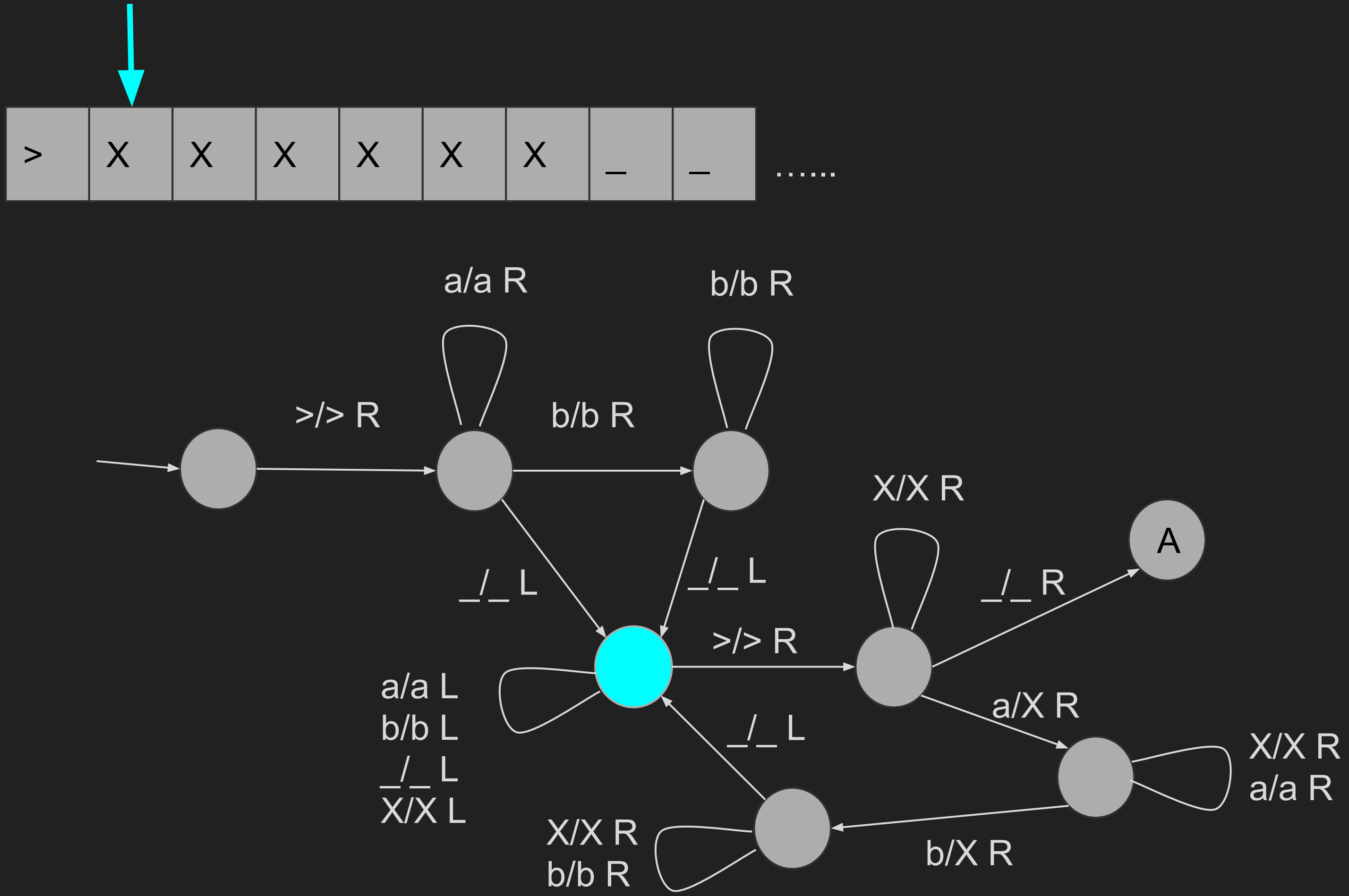
>
X
X
X
X
X
X
_
_
……
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



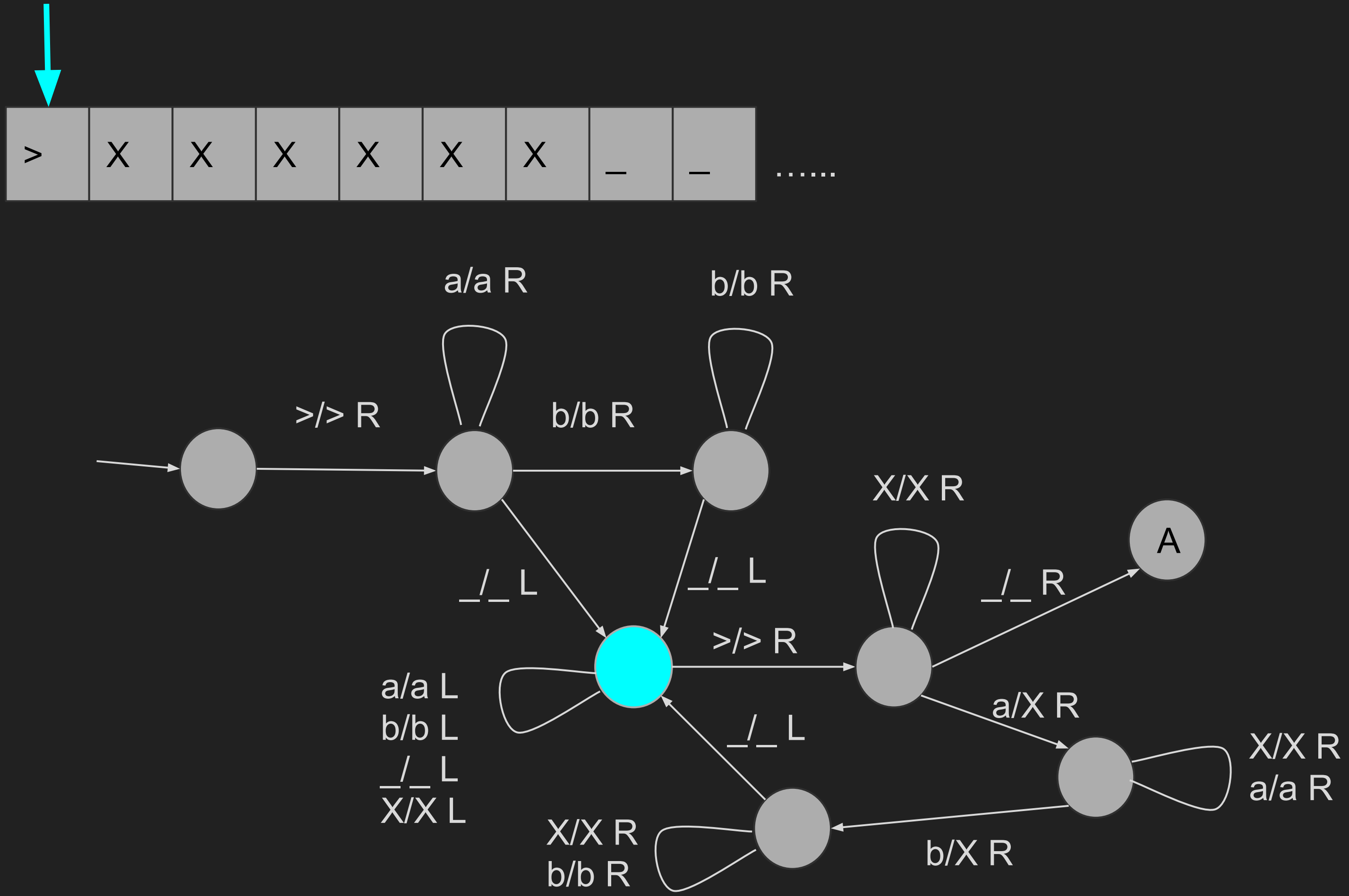
>
X
X
X
X
X
X
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
_/_ R
A
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



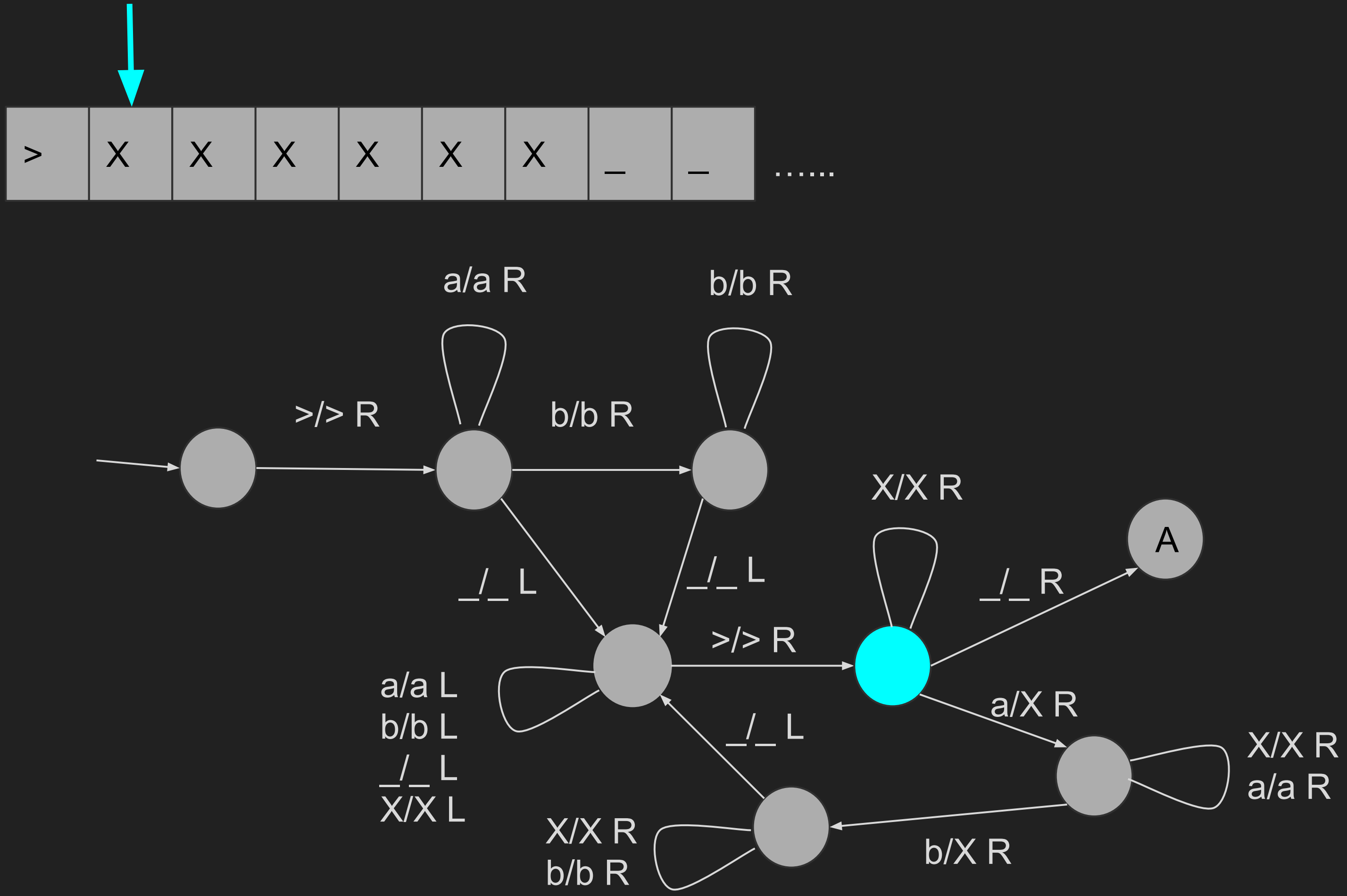
>
X
X
X
X
X
X
_
_
……
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

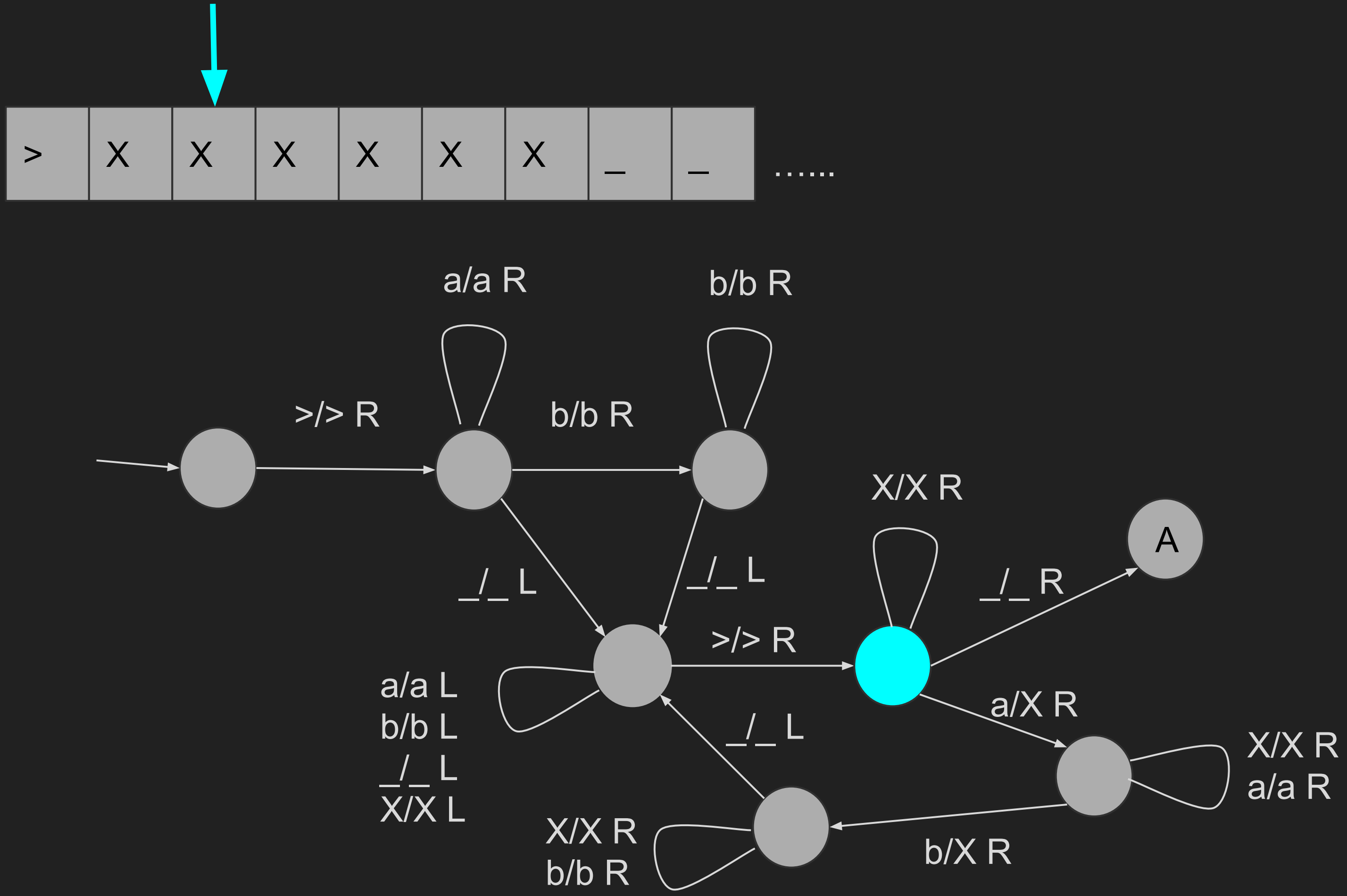


>
X
X
X
X
X
X
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

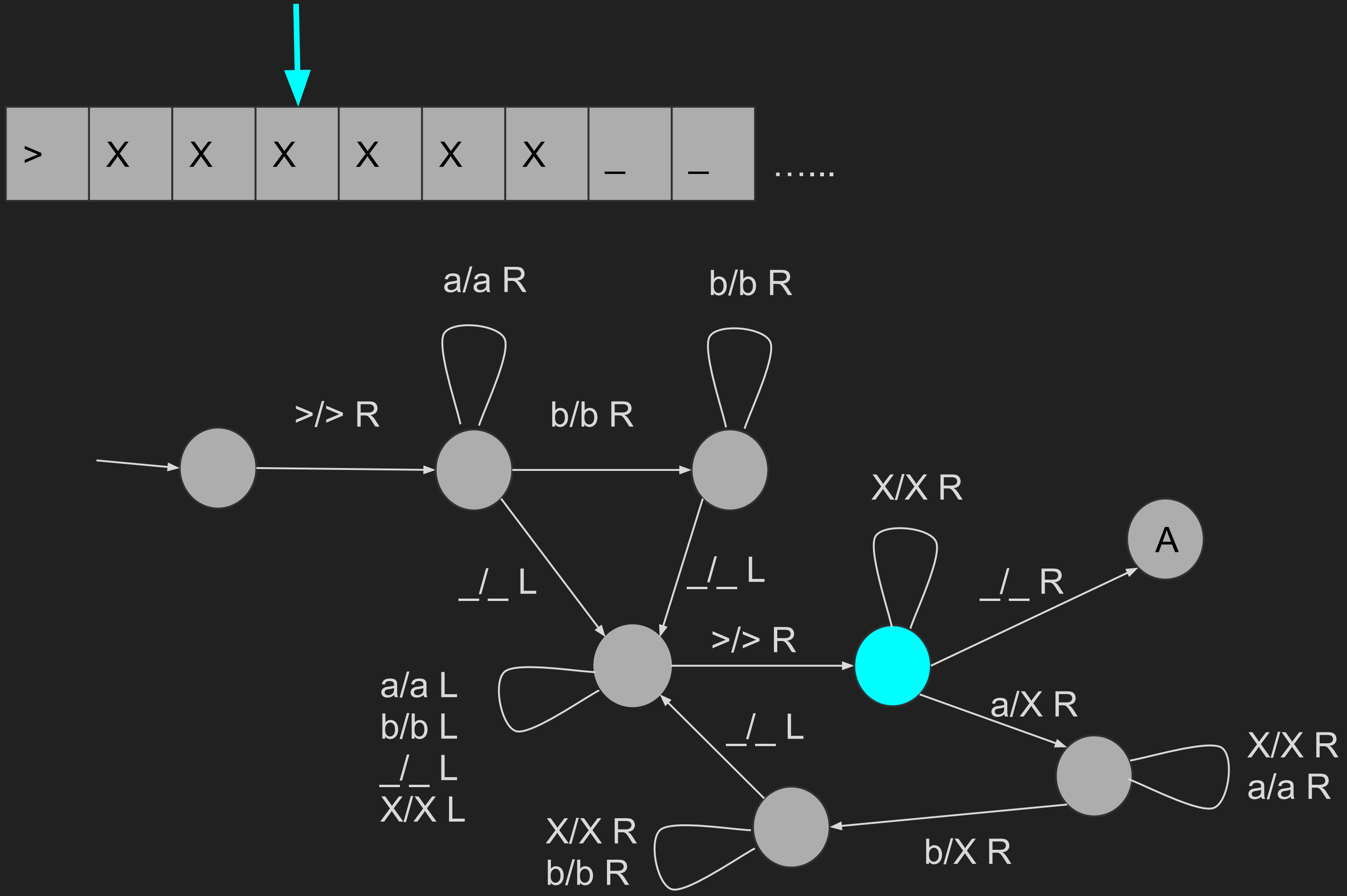


> X X X X X X _ _ ......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R

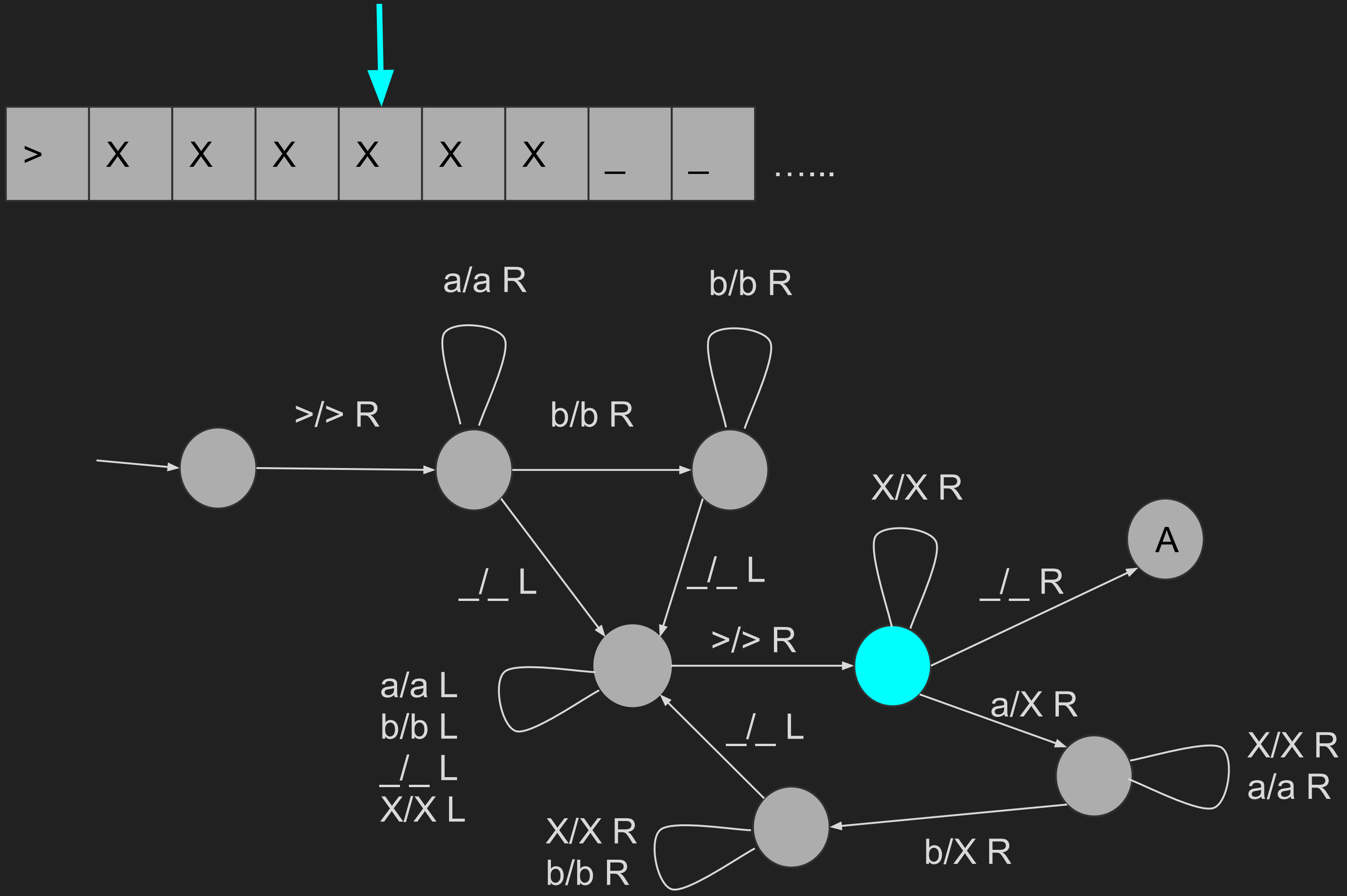

>
X X X X X X _ _
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
a/a L
b/b L
_/_ L
X/X L
>/> R
X/X R
_/_ R
A
a/X R
X/X R
a/a R
b/X R
_/_ L
X/X R
b/b R



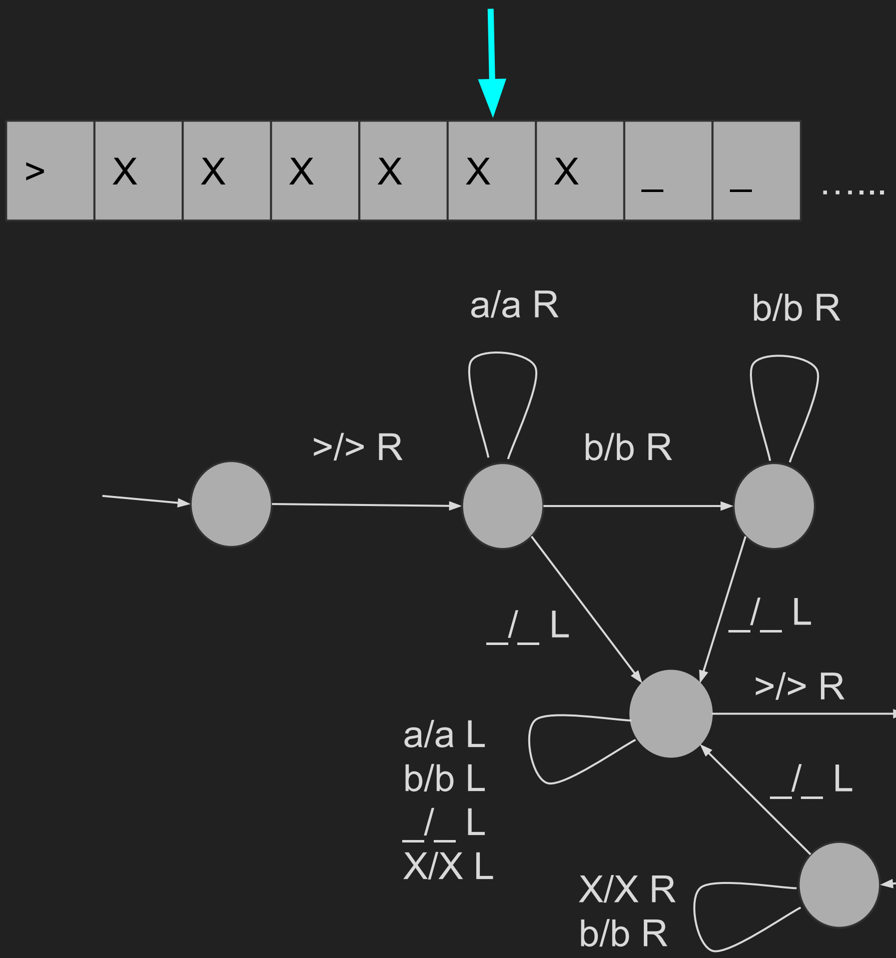
> X X X X X X _ _ ......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



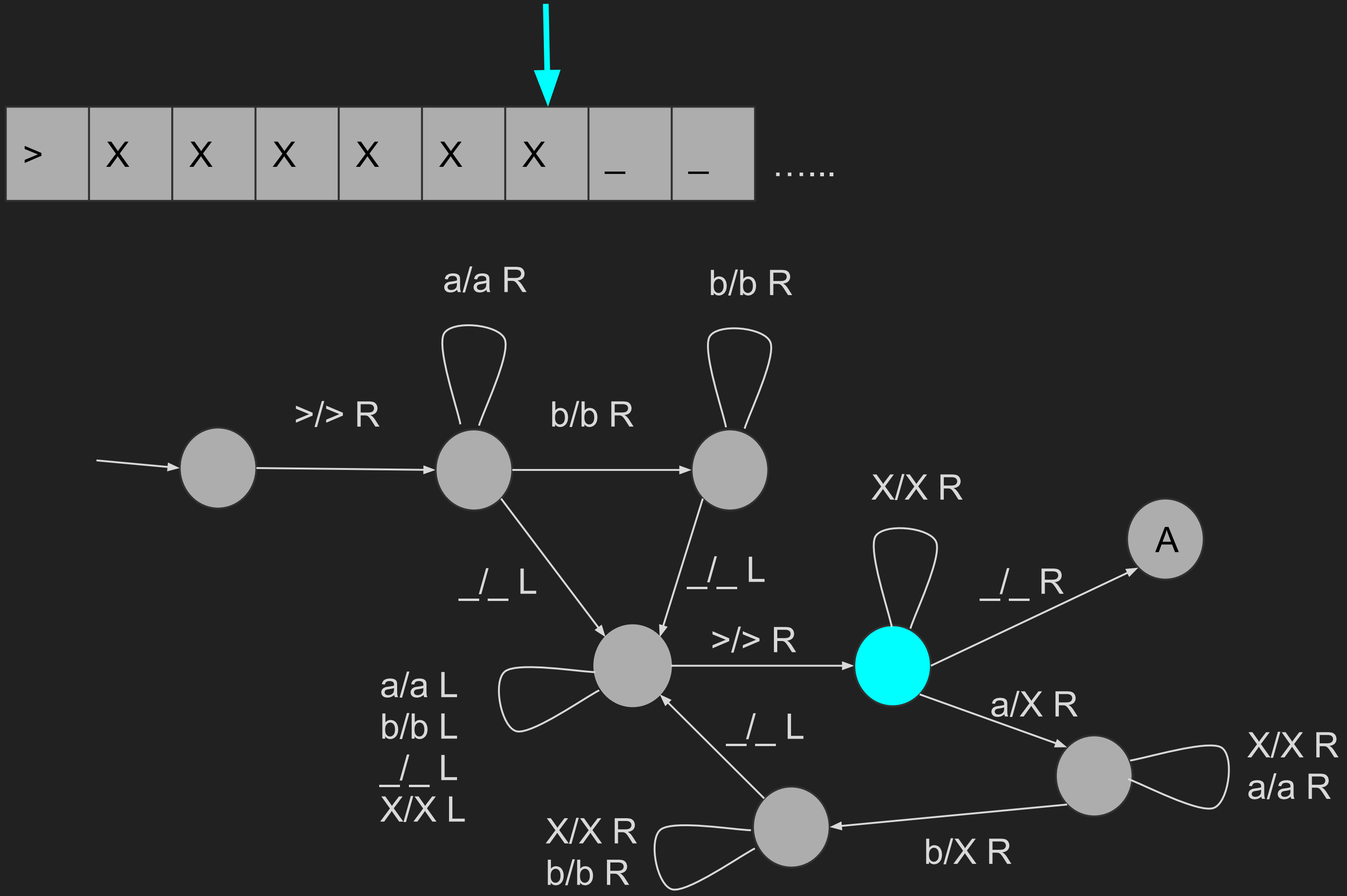
>
X
X
X
X
X
X
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



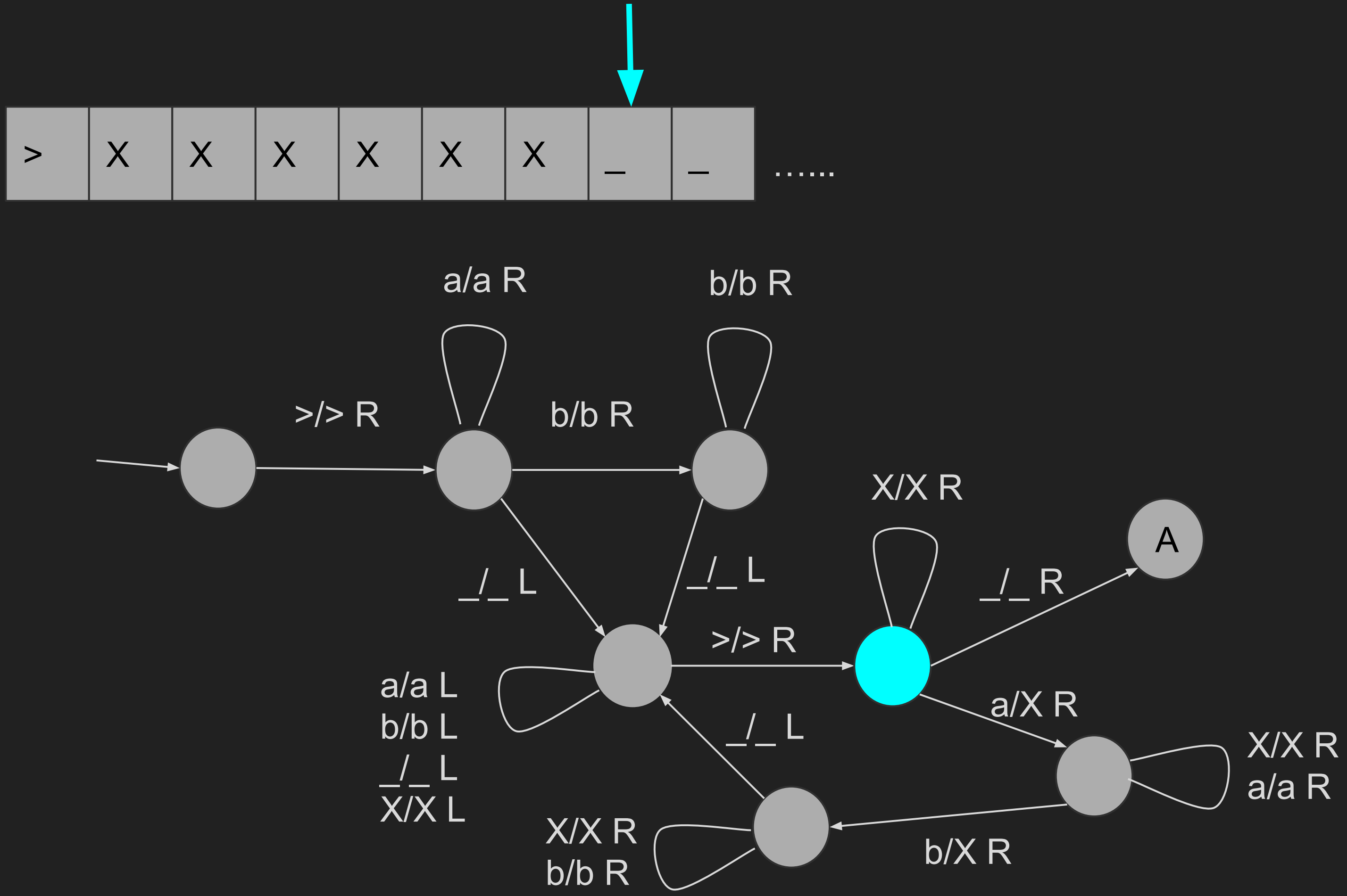
>
X
X
X
X
X
X
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



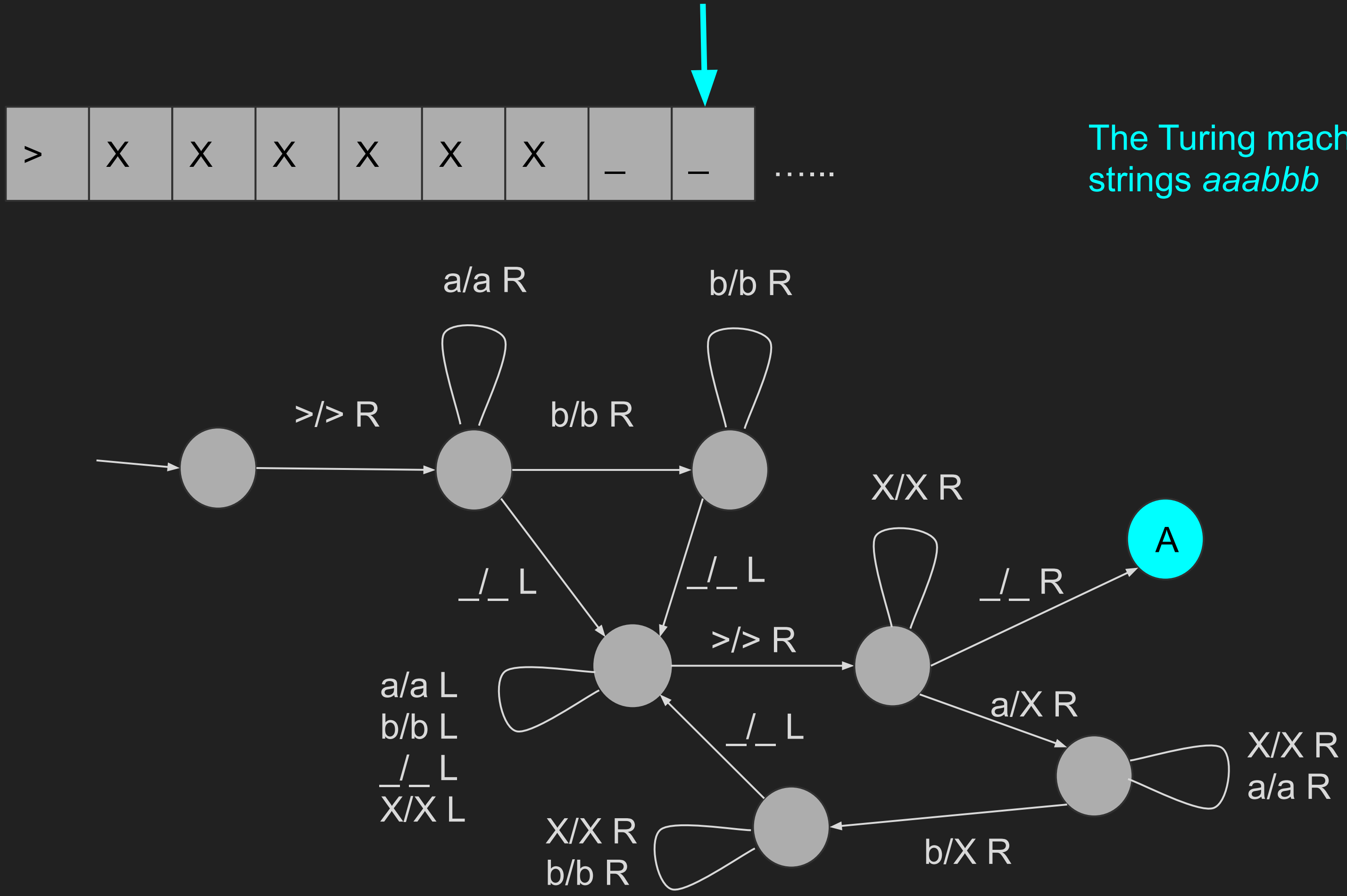
>
X
X
X
X
X
X
_
_
......
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
>/> R
_/_ R
A
a/a L
b/b L
_/_ L
X/X L
a/X R
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



> X X X X X X _ _ ……
>/> R
a/a R
b/b R
b/b R
_/_ L
_/_ L
X/X R
A
_/_ R
>/> R
a/a L
b/b L
_/_ L
X/X L
a/X R
X/X R
a/a R
_/_ L
b/X R
X/X R
b/b R



>
X
X
X
X
X
X
_
_
......
a/a R
b/b R
>/> R
b/b R
X/X R
A
_/_ L
_/_ L
_/_ R
>/> R
a/X R
a/a L
b/b L
_/_ L
X/X L
_/_ L
X/X R
a/a R
X/X R
b/b R
b/X R



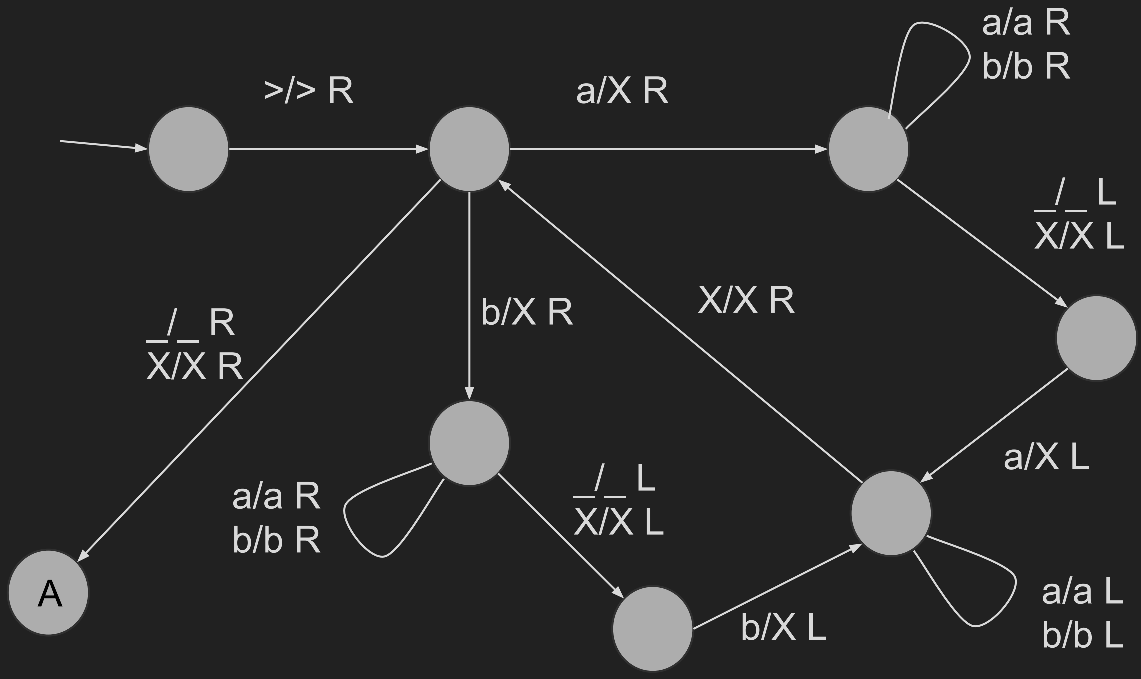
The Turing machine accepts strings *aaabbb*

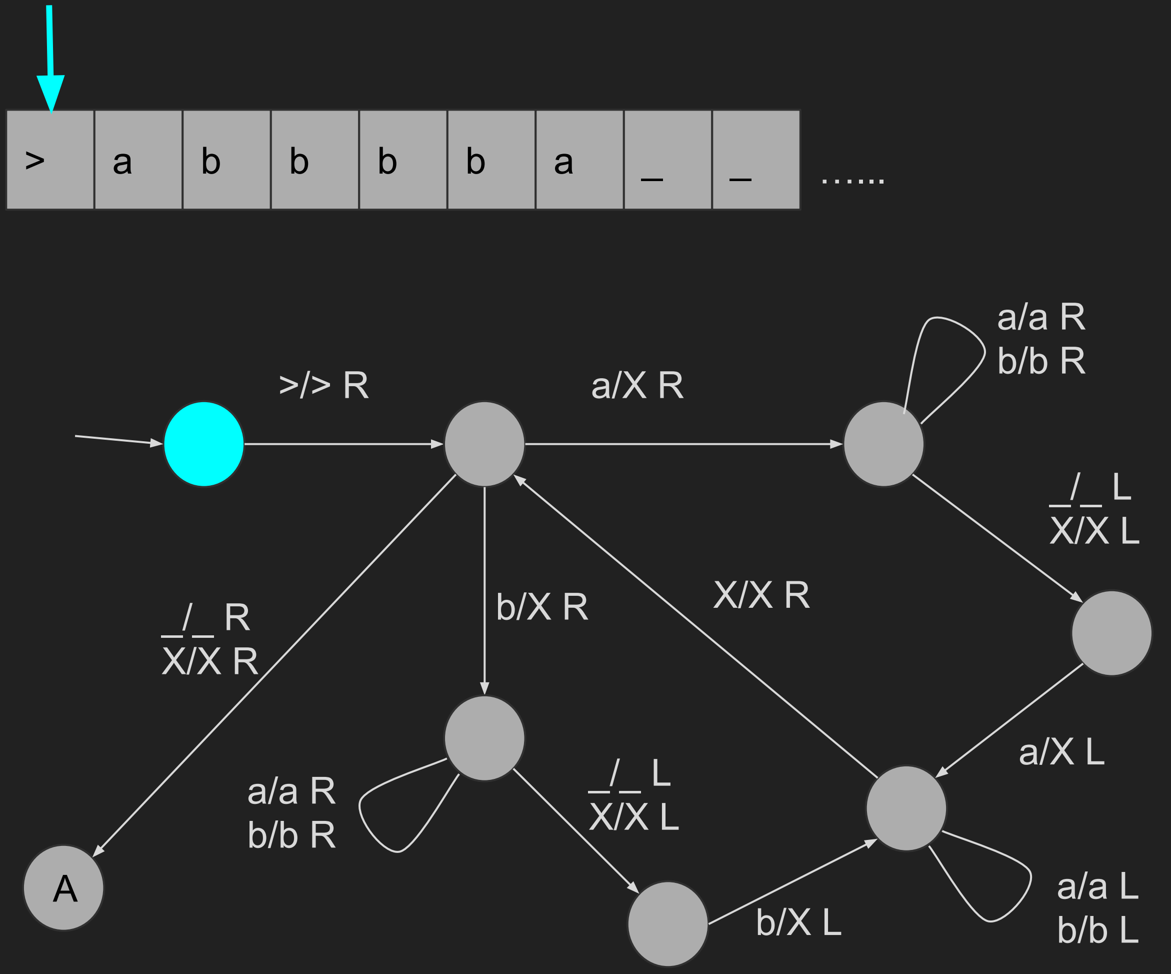
Example — $\{u \in \{a,b\}^* \mid u = rev(u)$, u even length$\}$

Intuition:

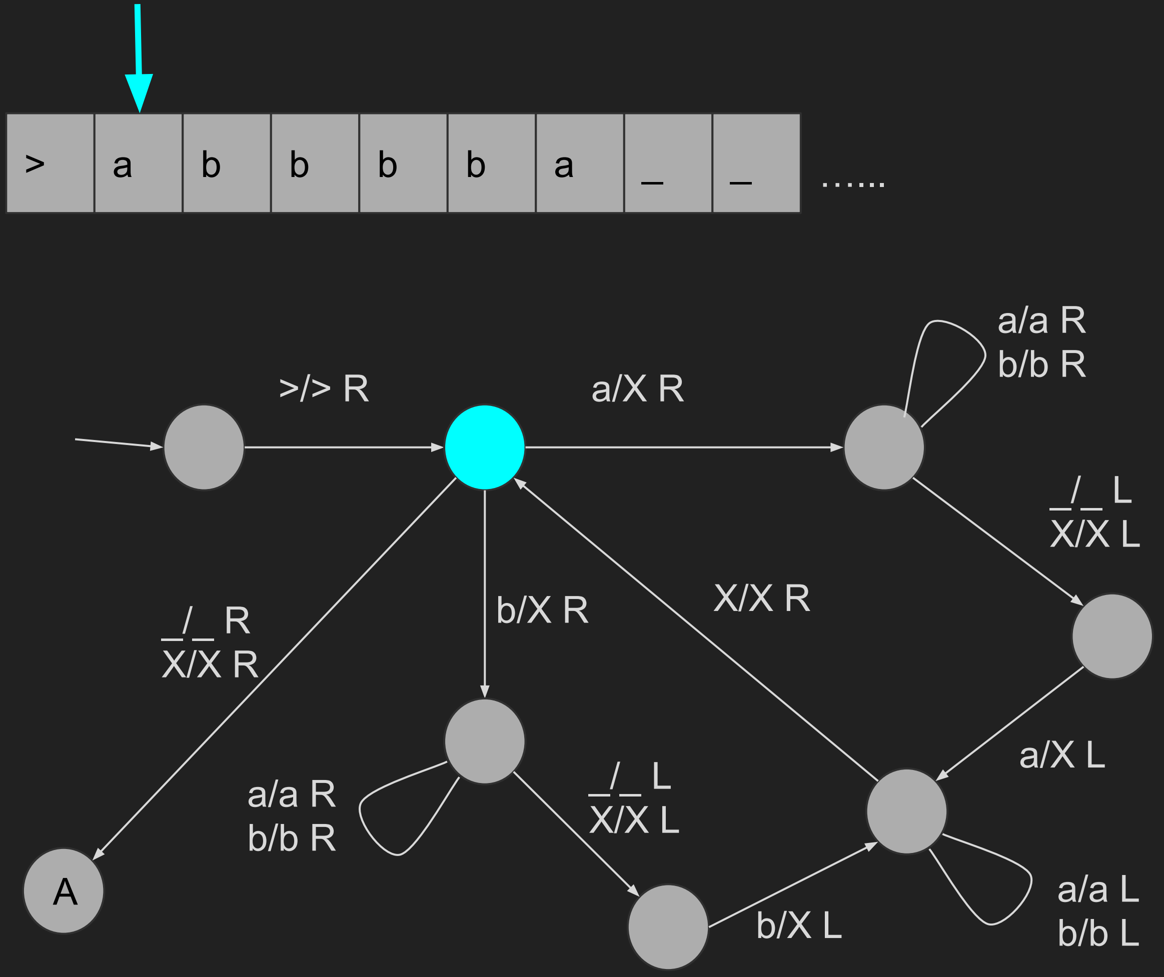
1. scan from left to right, finding the first uncrossed symbol
   a. if there is none, accept
2. if it's an a: cross it, go to the end of the tape, match an a there and cross it
   a. if the last uncrossed symbol is not an a, reject
3. if it's a b: cross it, go to the end of the tape, match a b there and cross it
   a. if the last uncrossed symbol is not a b, reject
4. rewind back to a crossed symbol and go to step 1

Example — {u ∈ {a,b}* | u = rev(u), u even length}

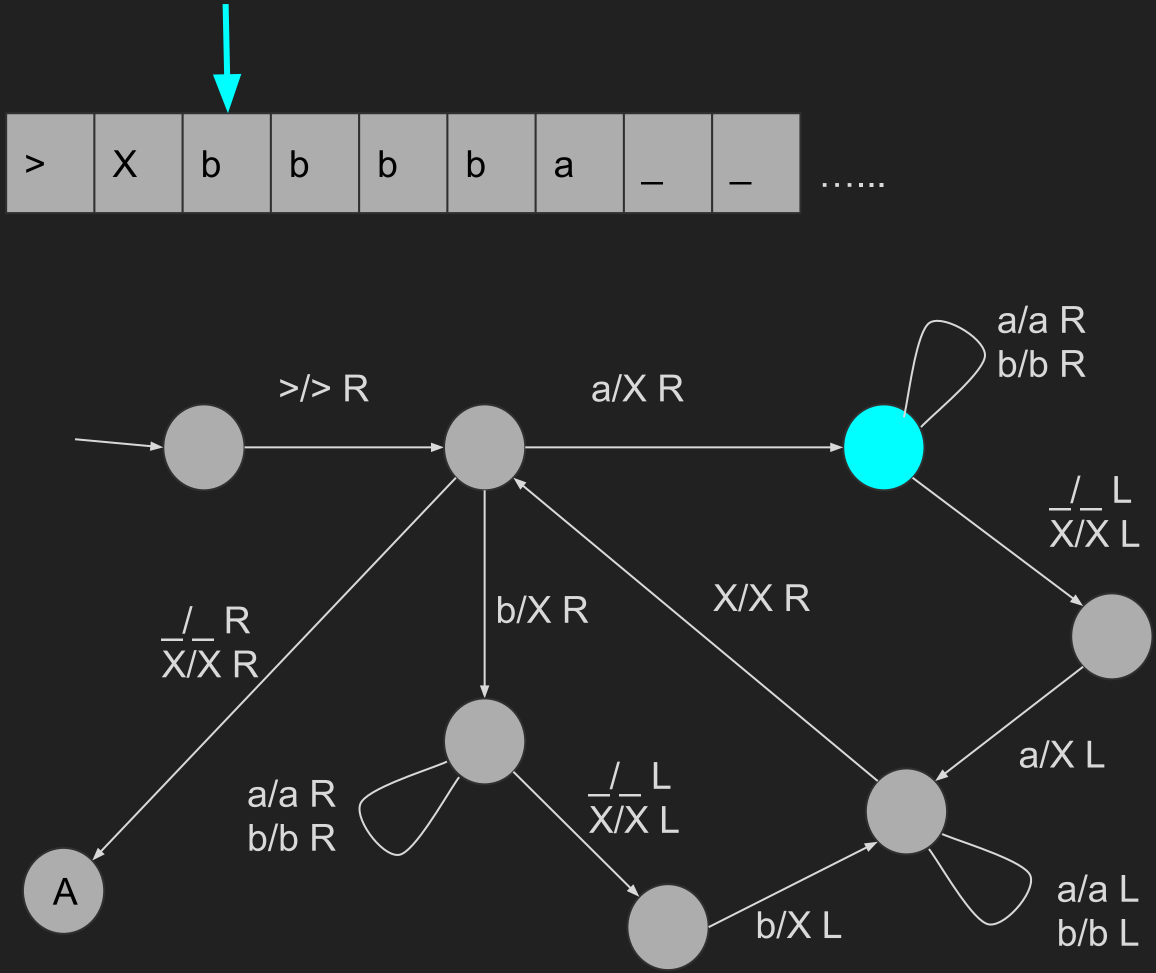




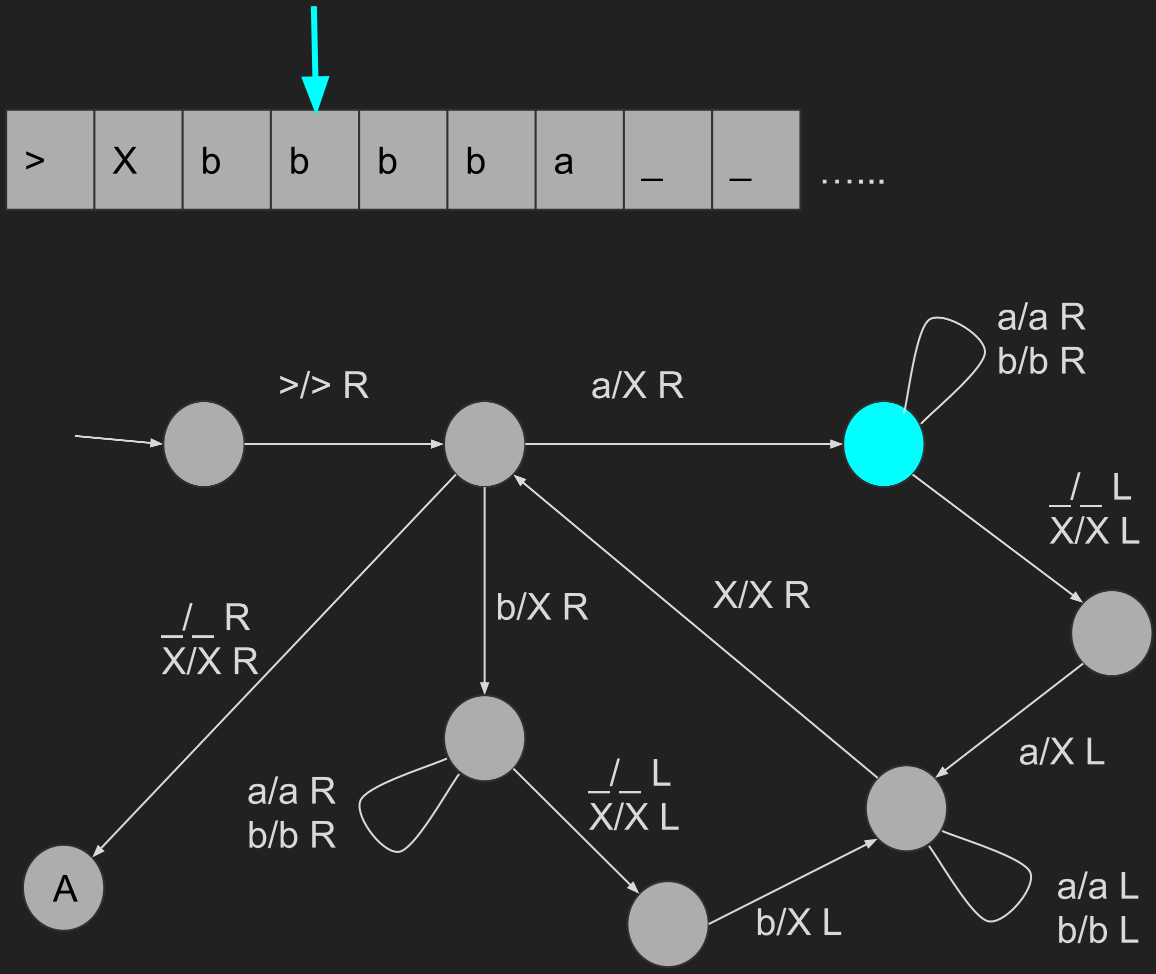
>
a
b
b
b
b
a
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



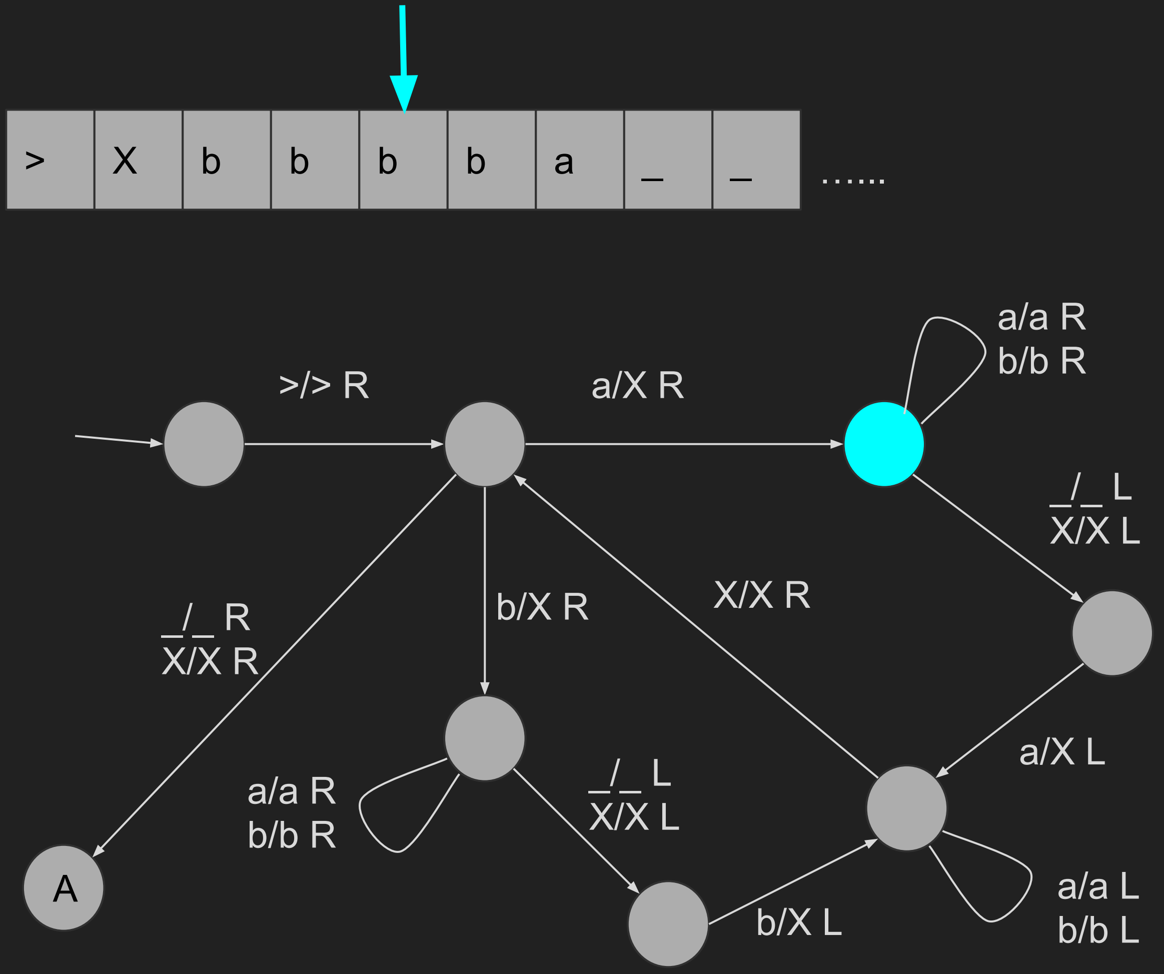
>
a
b
b
b
b
a
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
a/a L
b/b L
b/X L
A



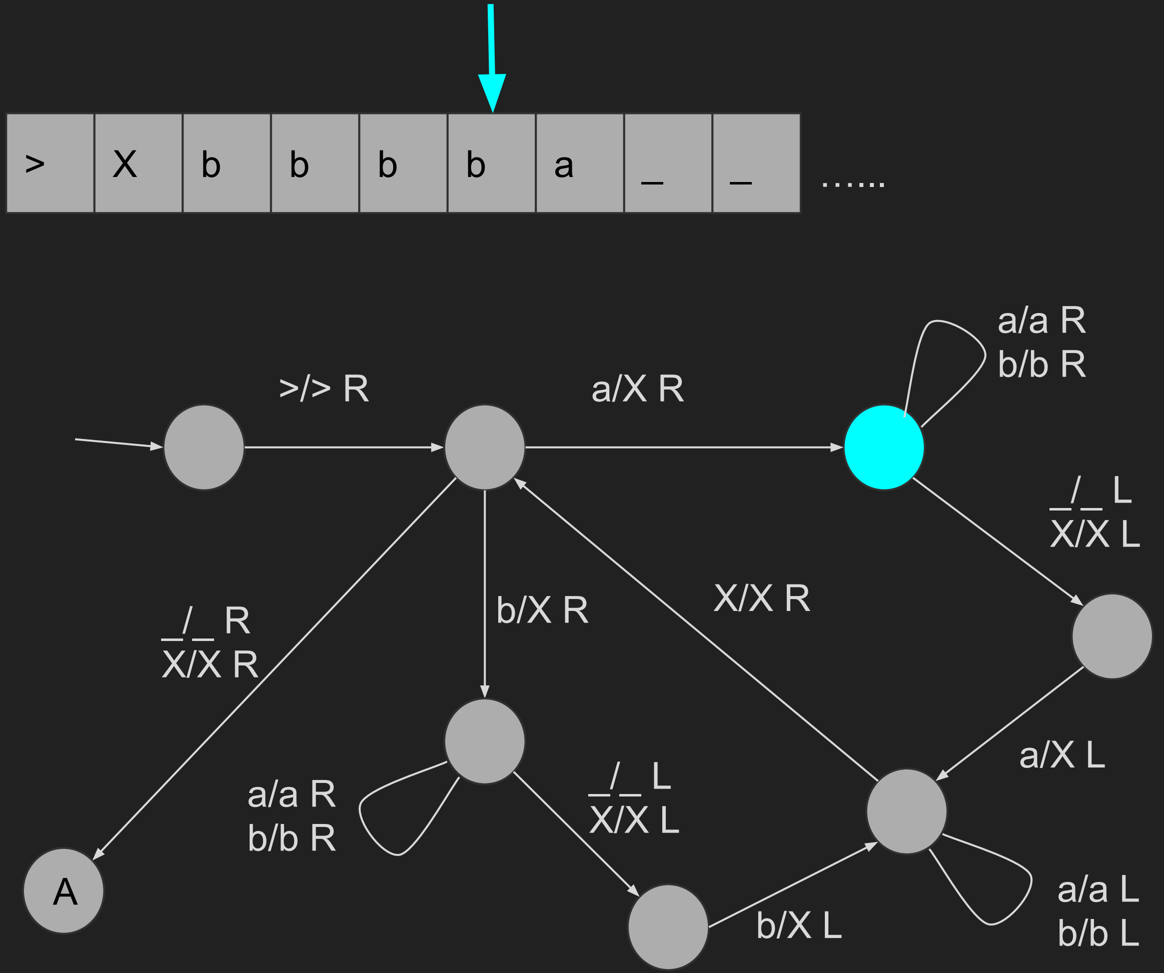
>
X
b
b
b
b
a
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
b/X R
X/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



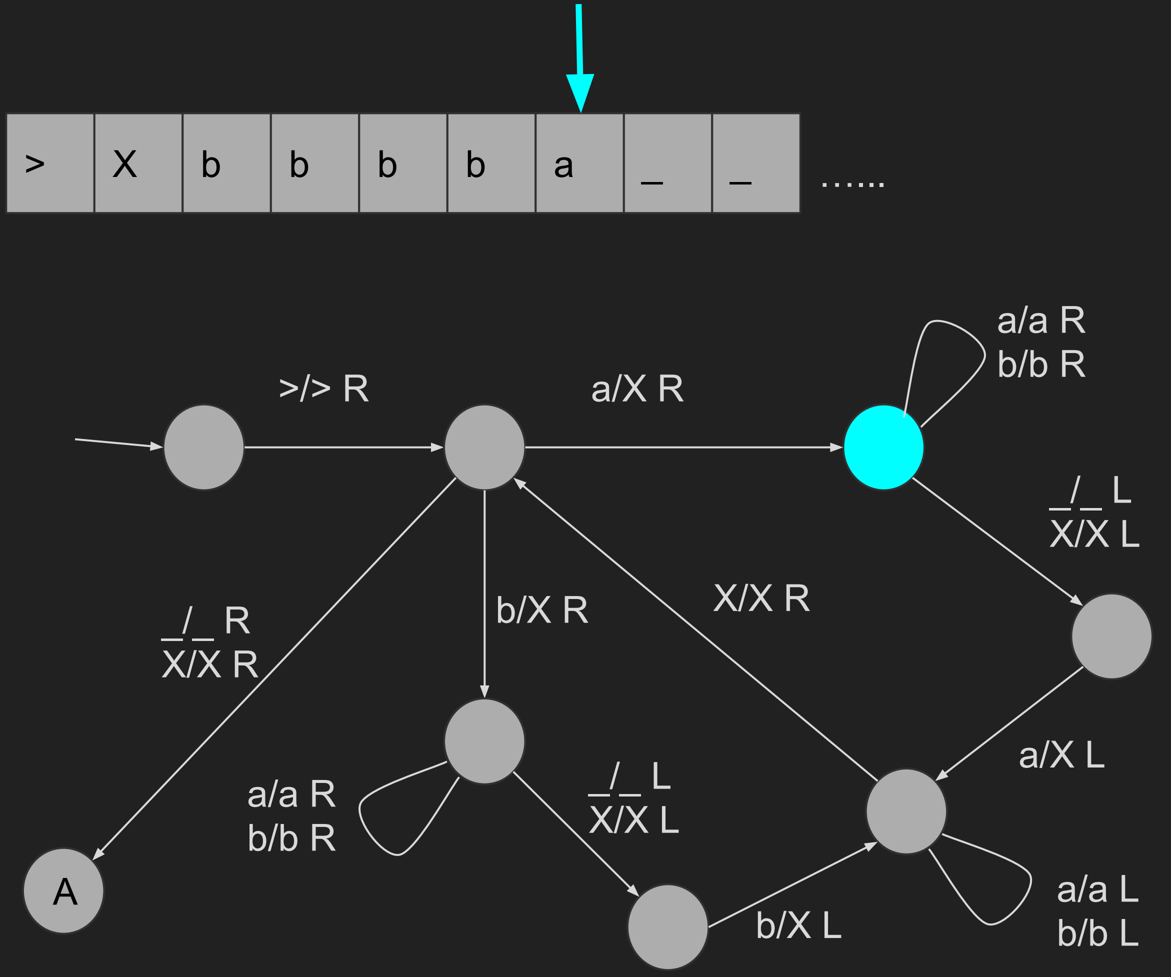
>
X
b
b
b
b
a
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



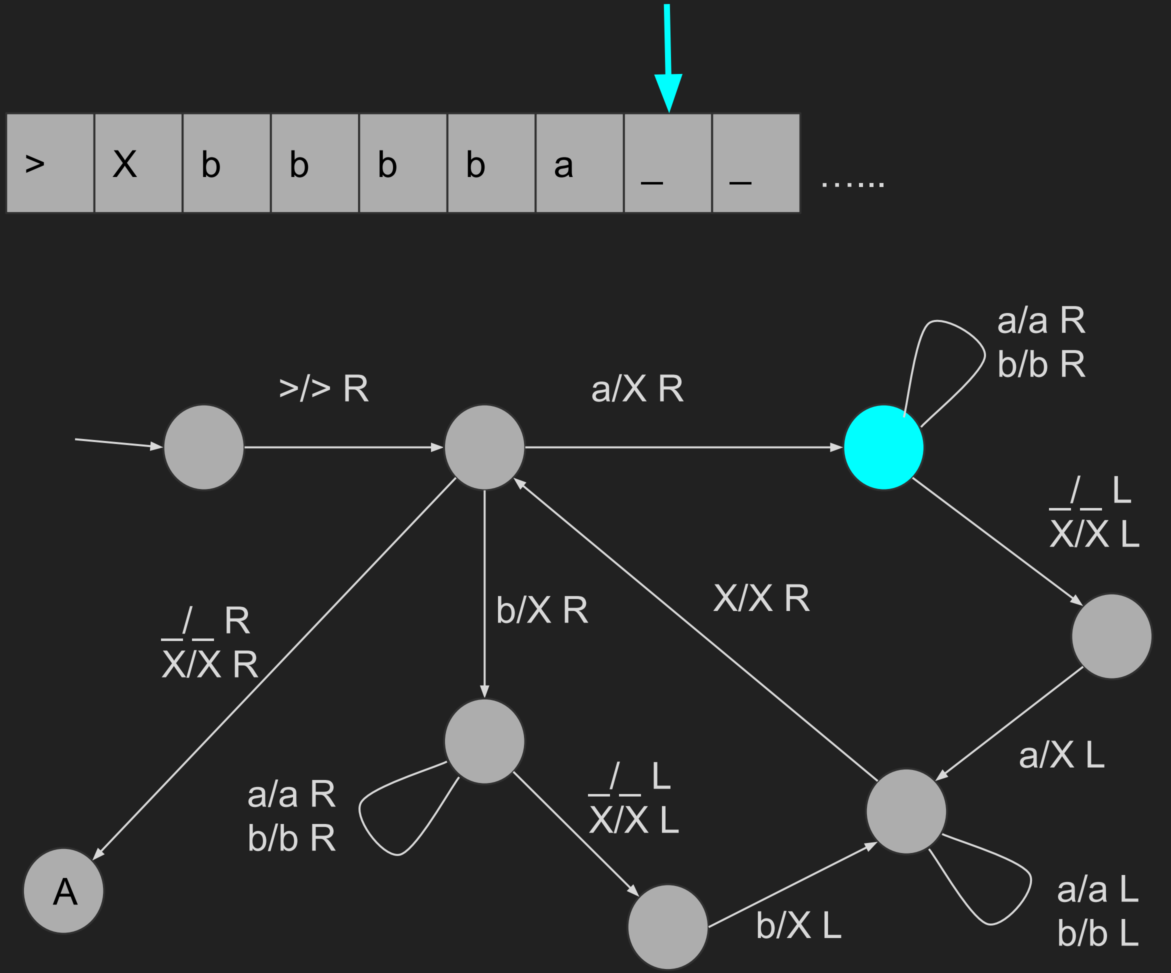
>
X
b
b
b
b
a
_
_
……
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



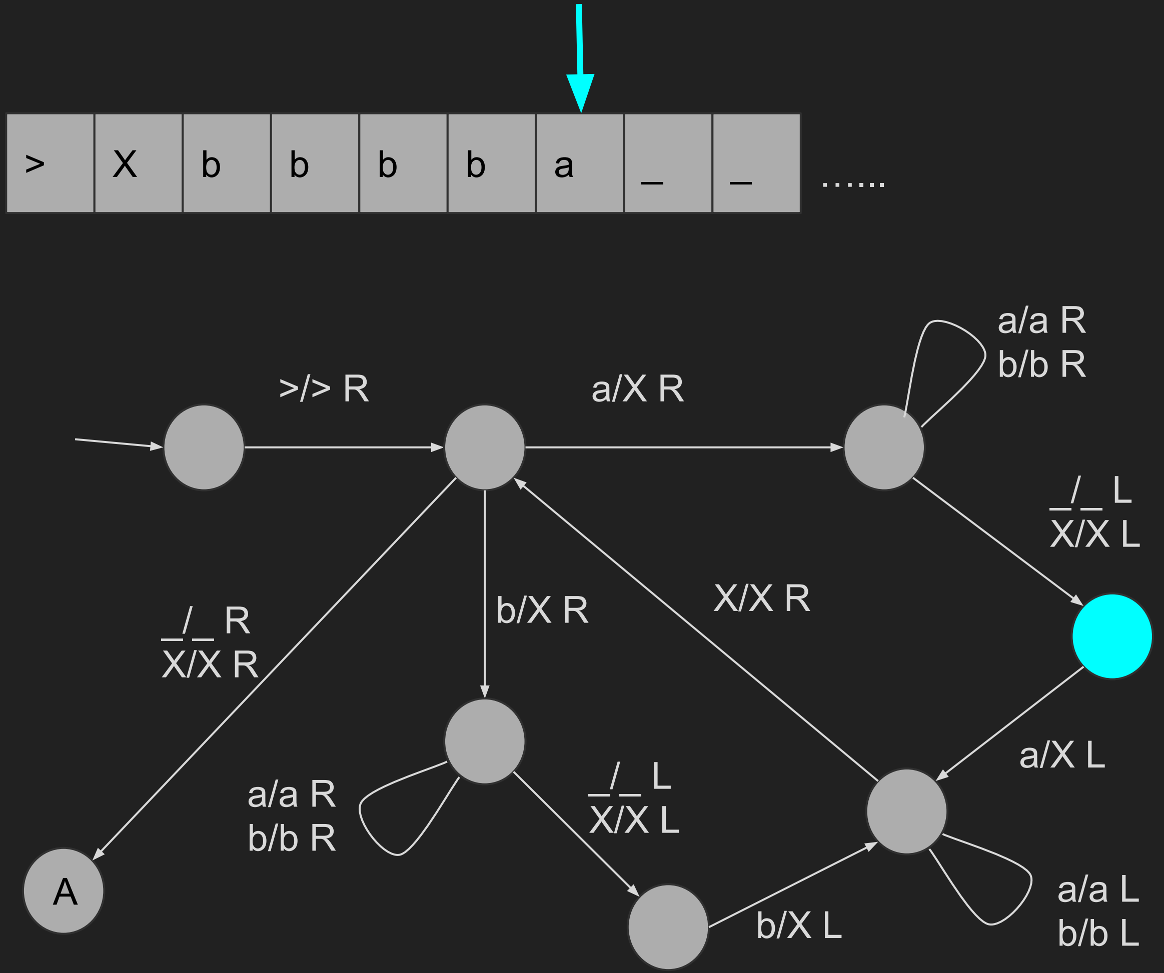
> X b b b b a _ _ ......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



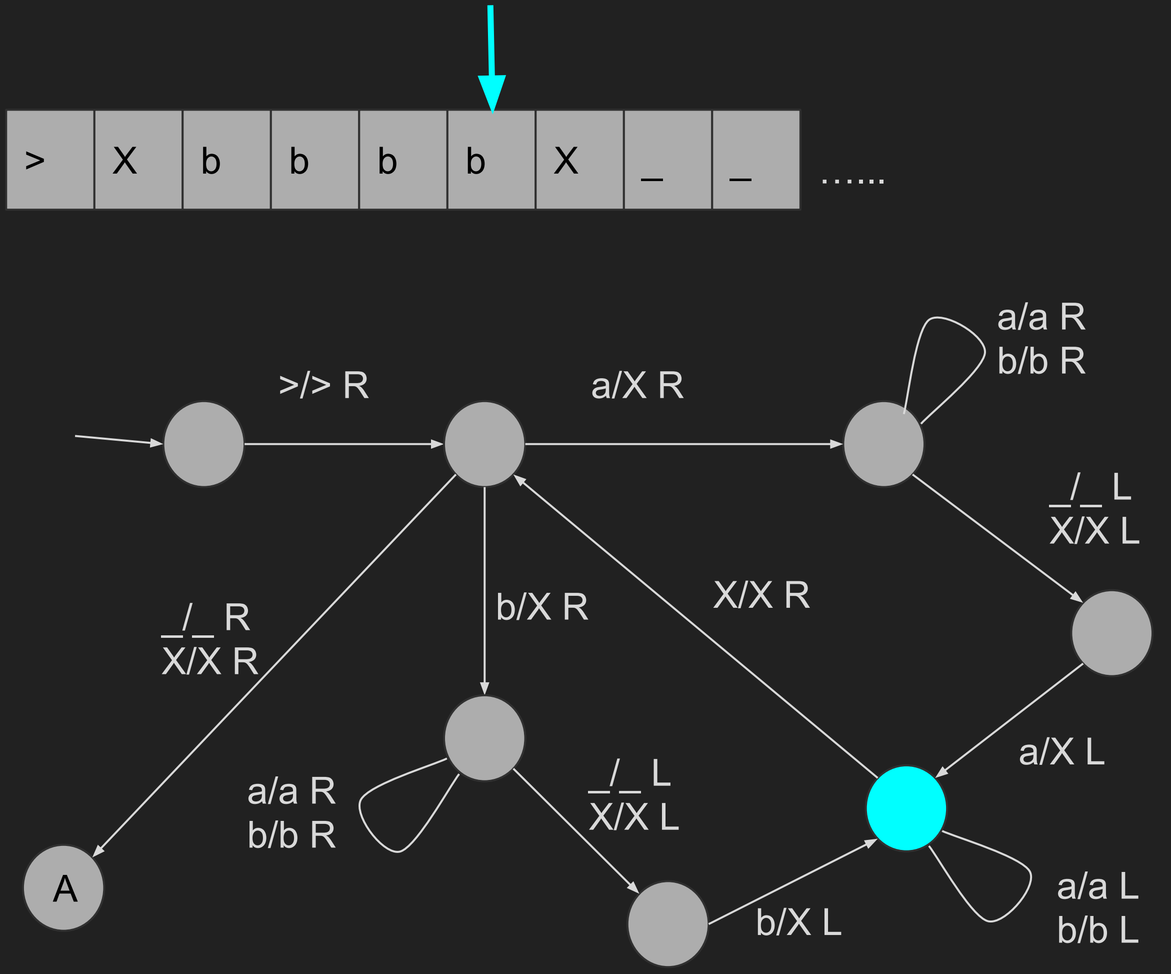
>
X
b
b
b
b
a
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
b/X R
X/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
a/a L
b/b L
b/X L



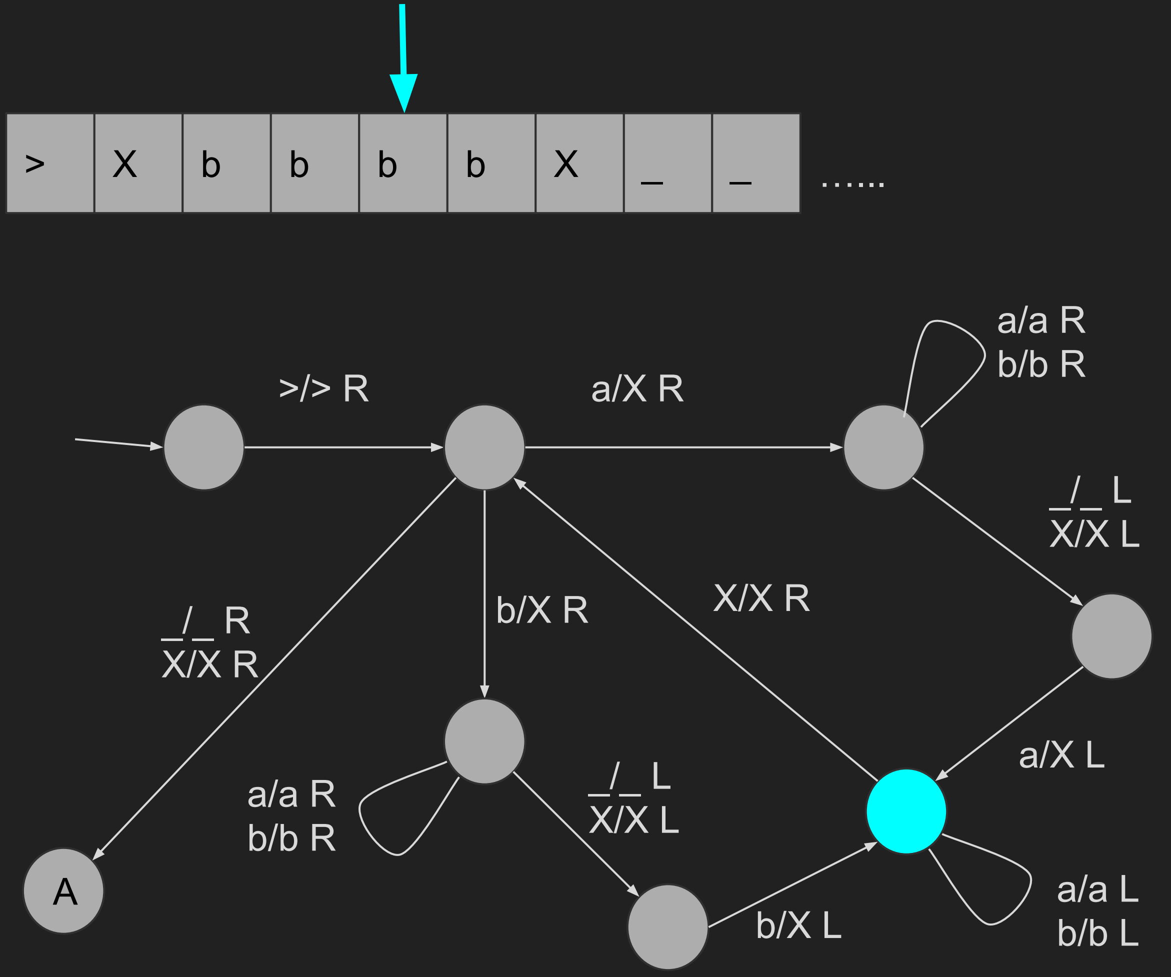
>
X
b
b
b
b
a
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L

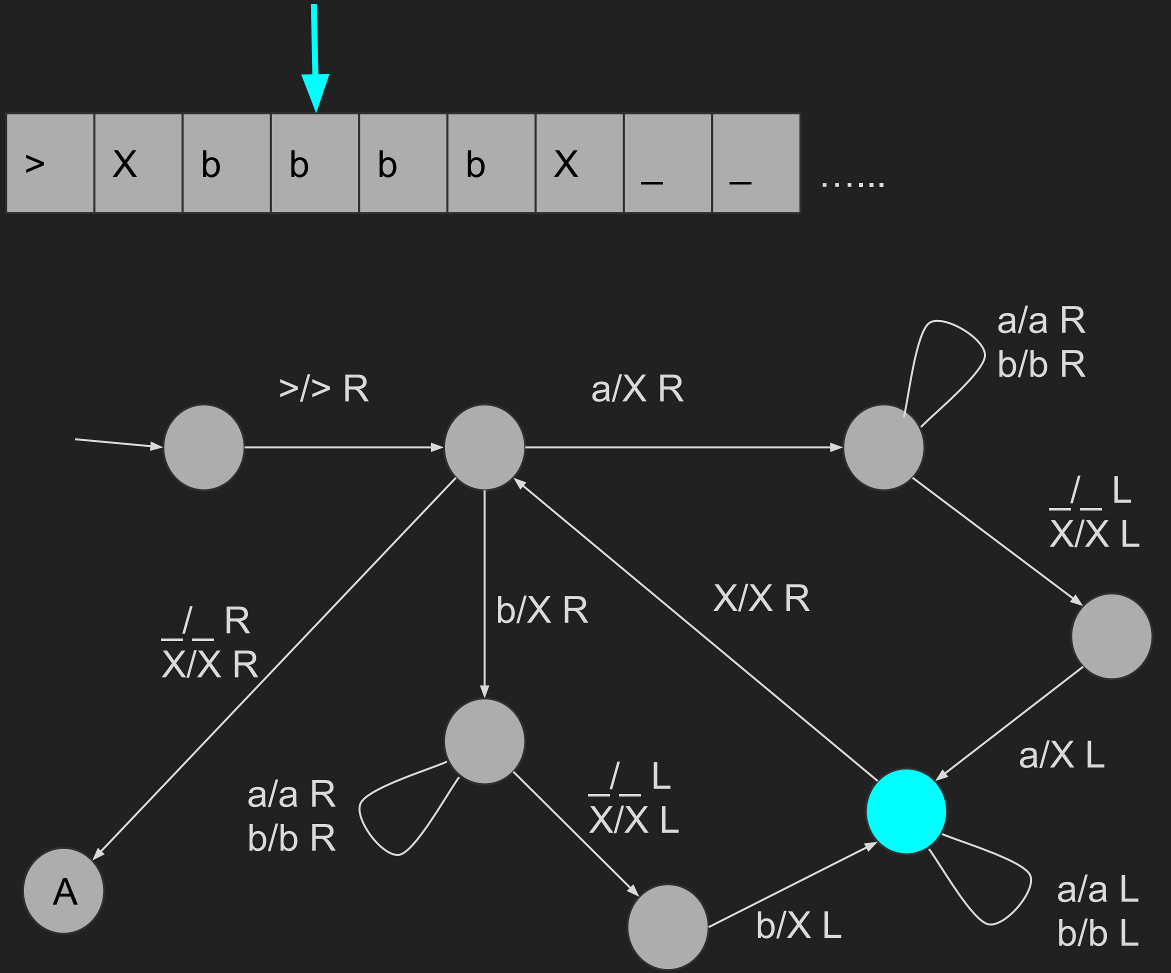


>
X
b
b
b
b
a
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L

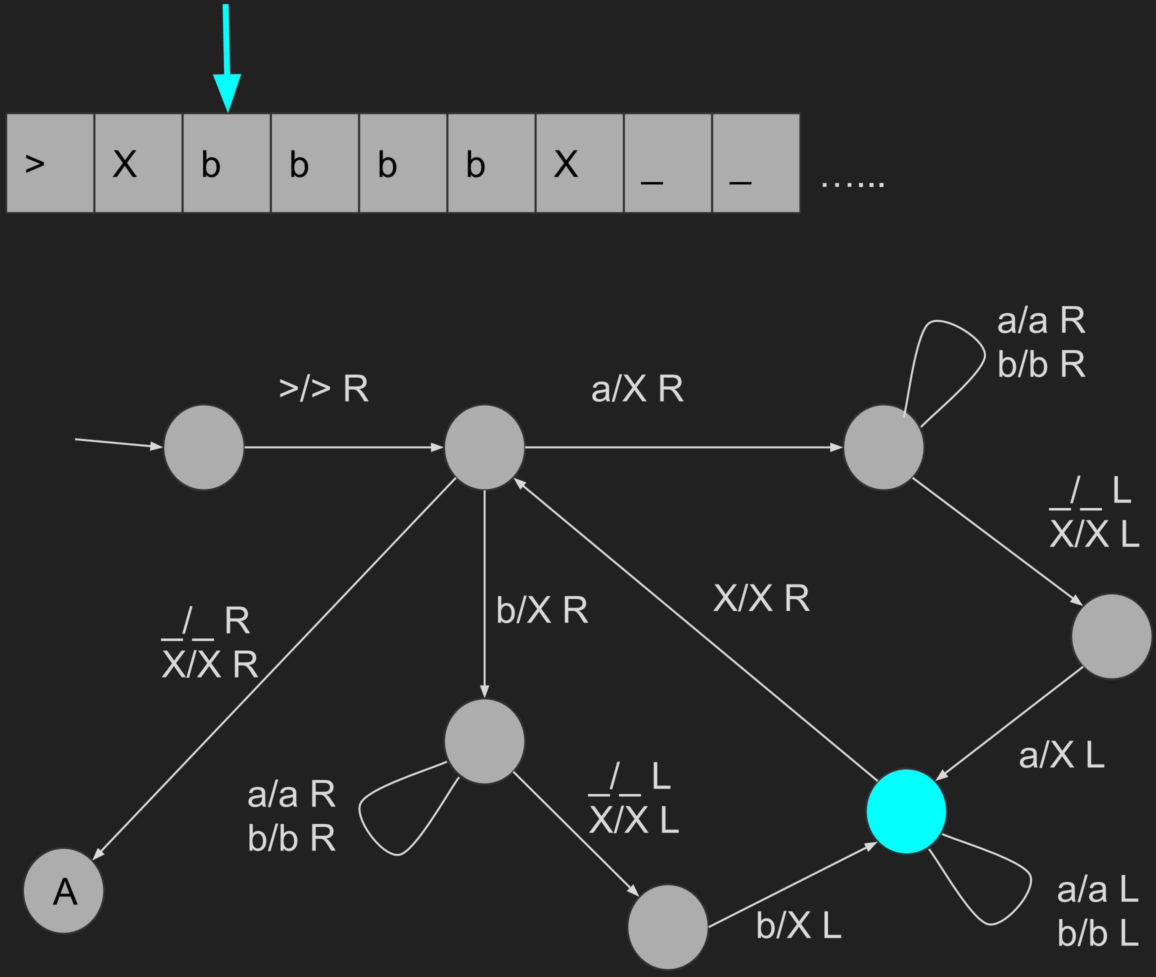


>
X
b
b
b
b
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L

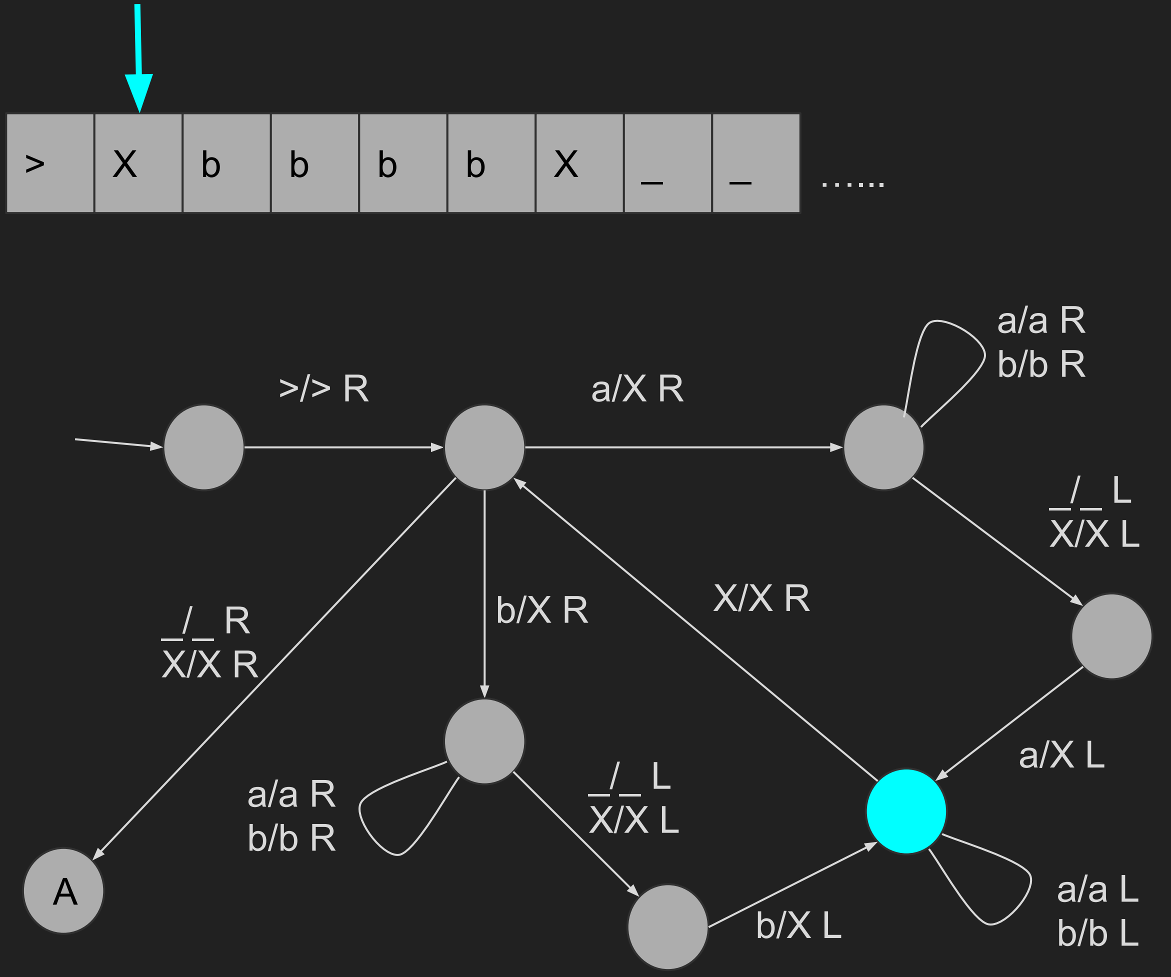

> X b b b b X _ _ ......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
b/X R
X/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



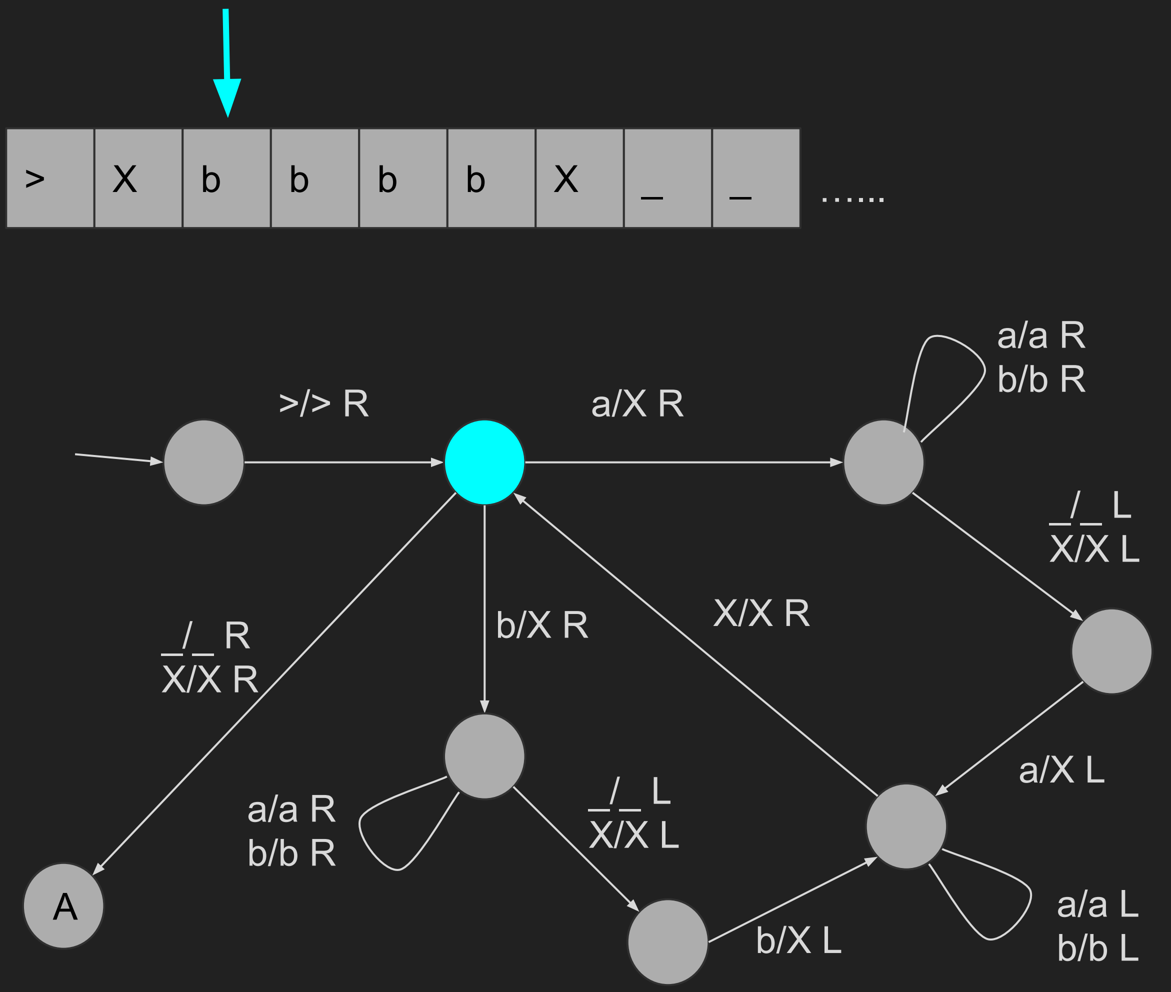
>
X
b
b
b
b
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



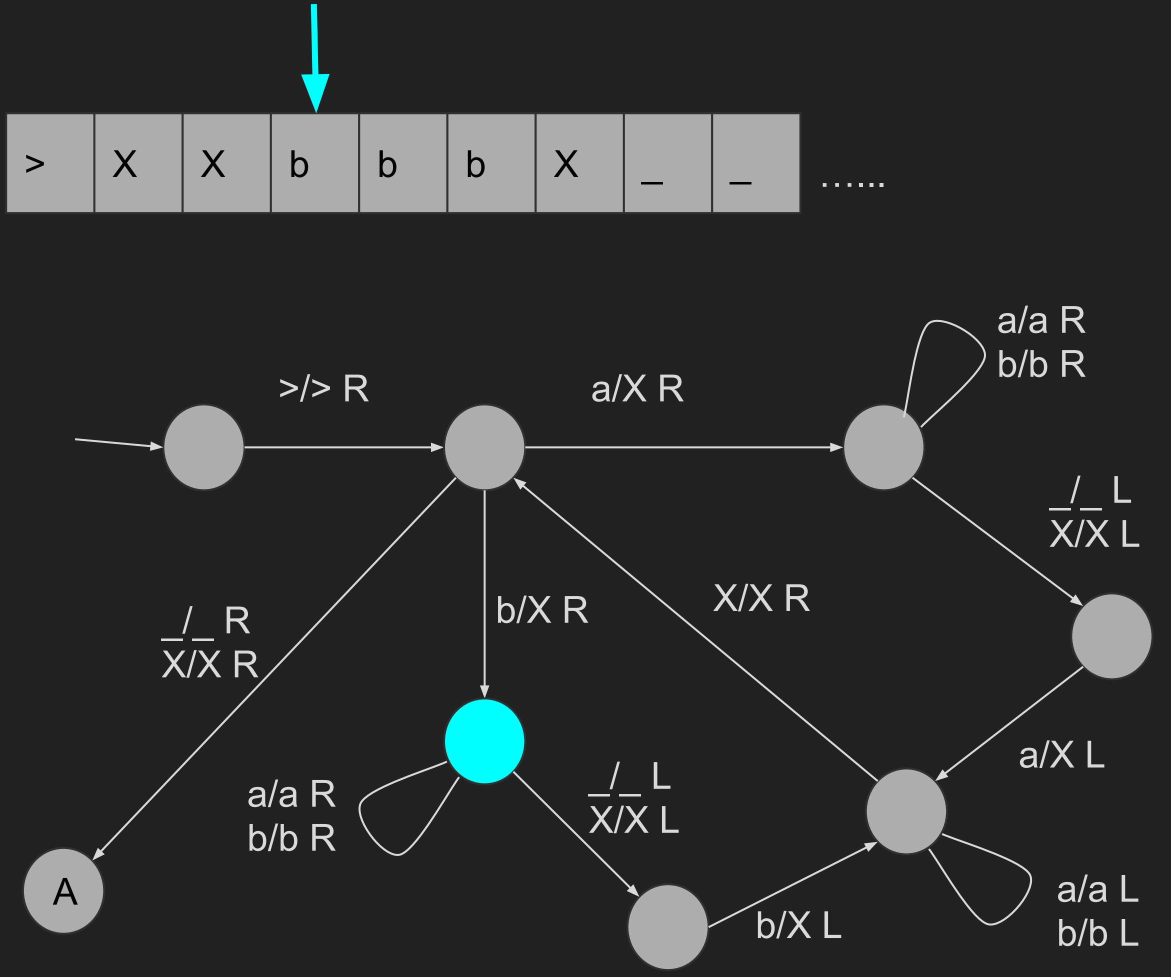
>
X
b
b
b
b
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



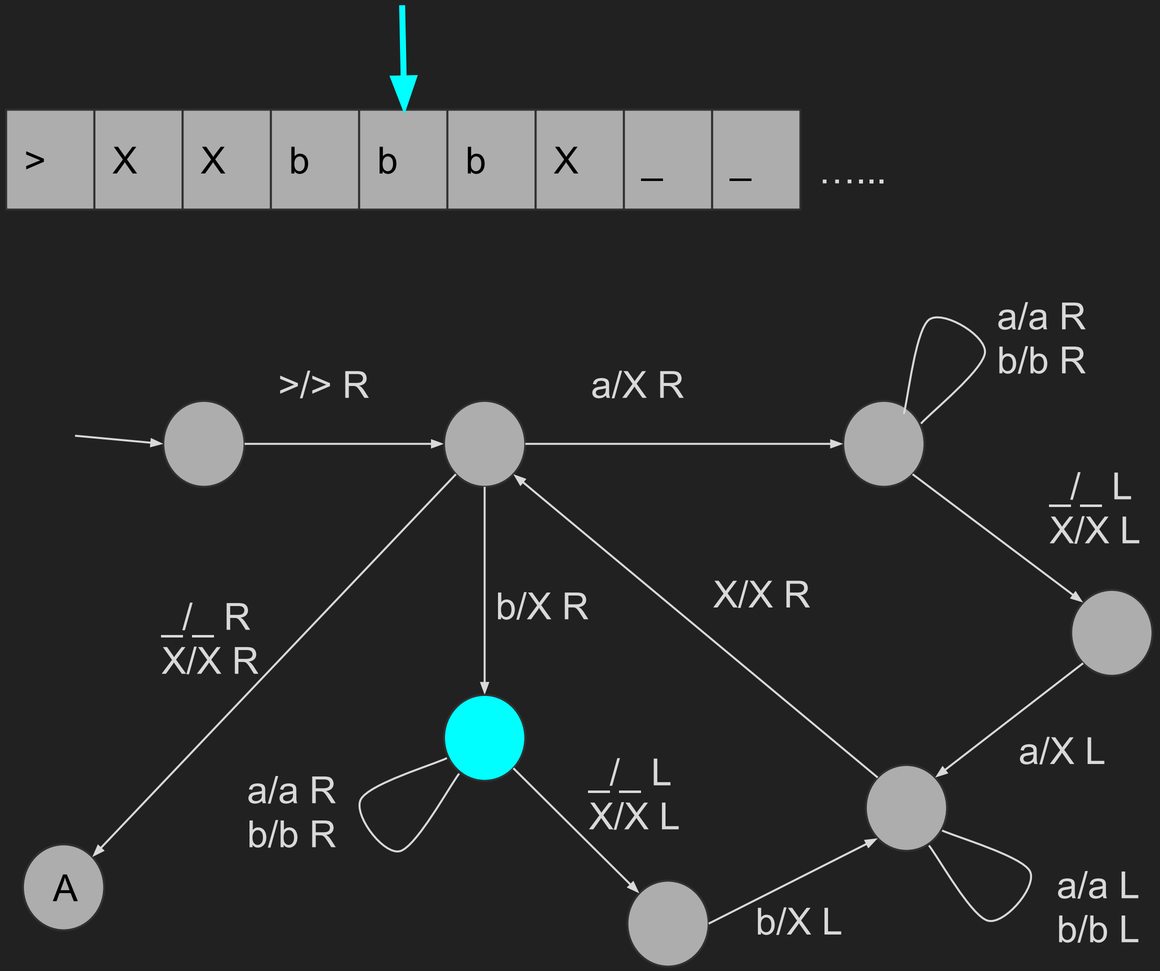
>
X
b
b
b
b
X
_
_
……
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



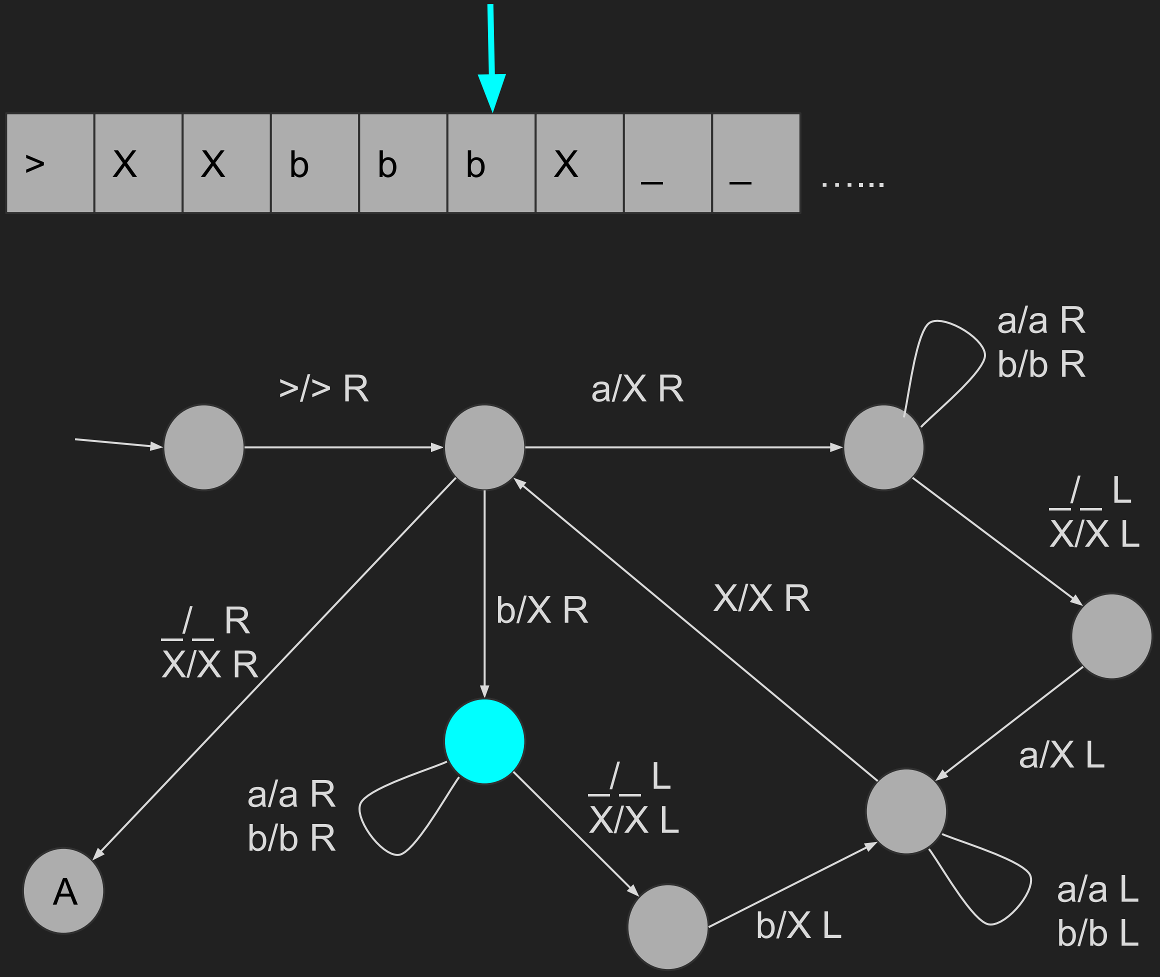
>
X
b
b
b
b
X
_
_
……
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
b/X R
X/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



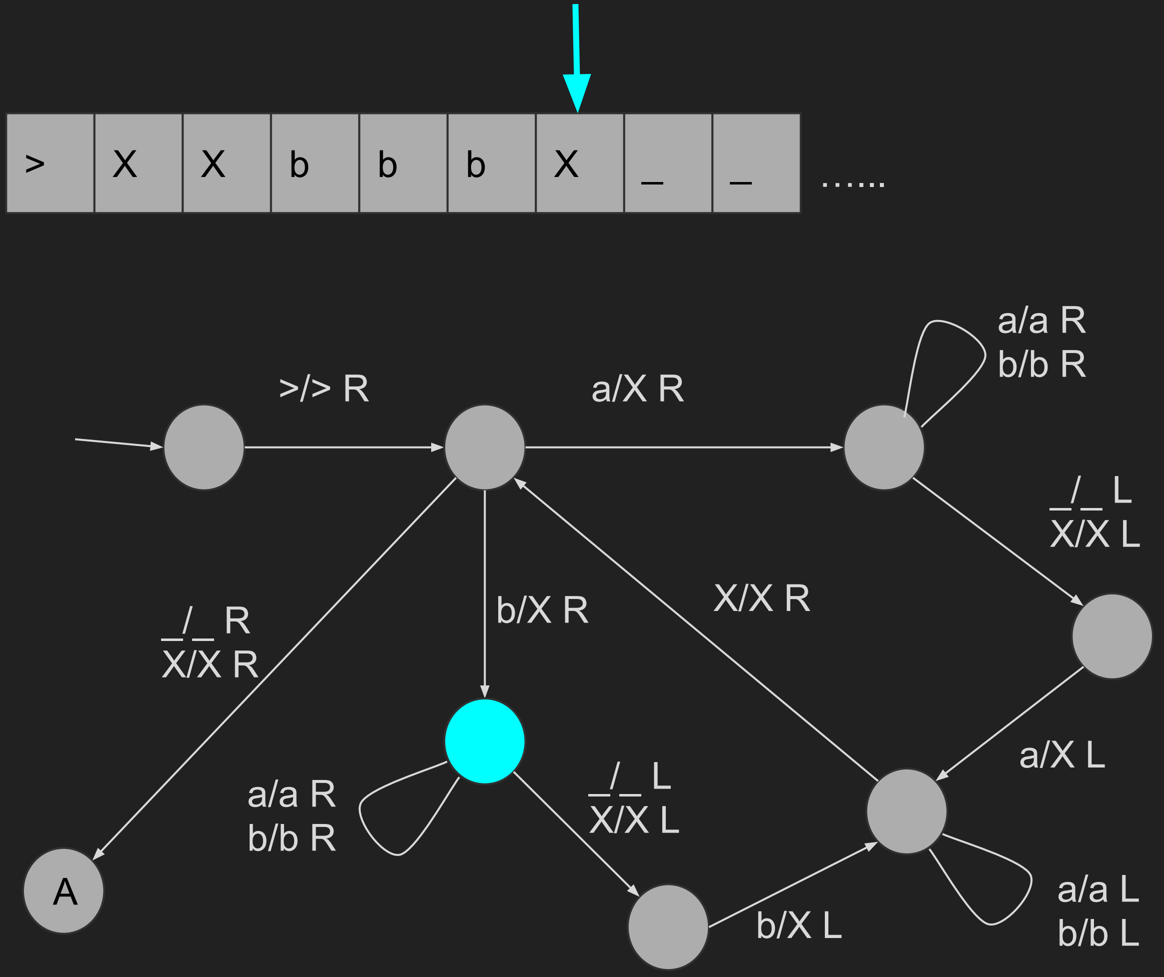
>
X
X
b
b
b
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



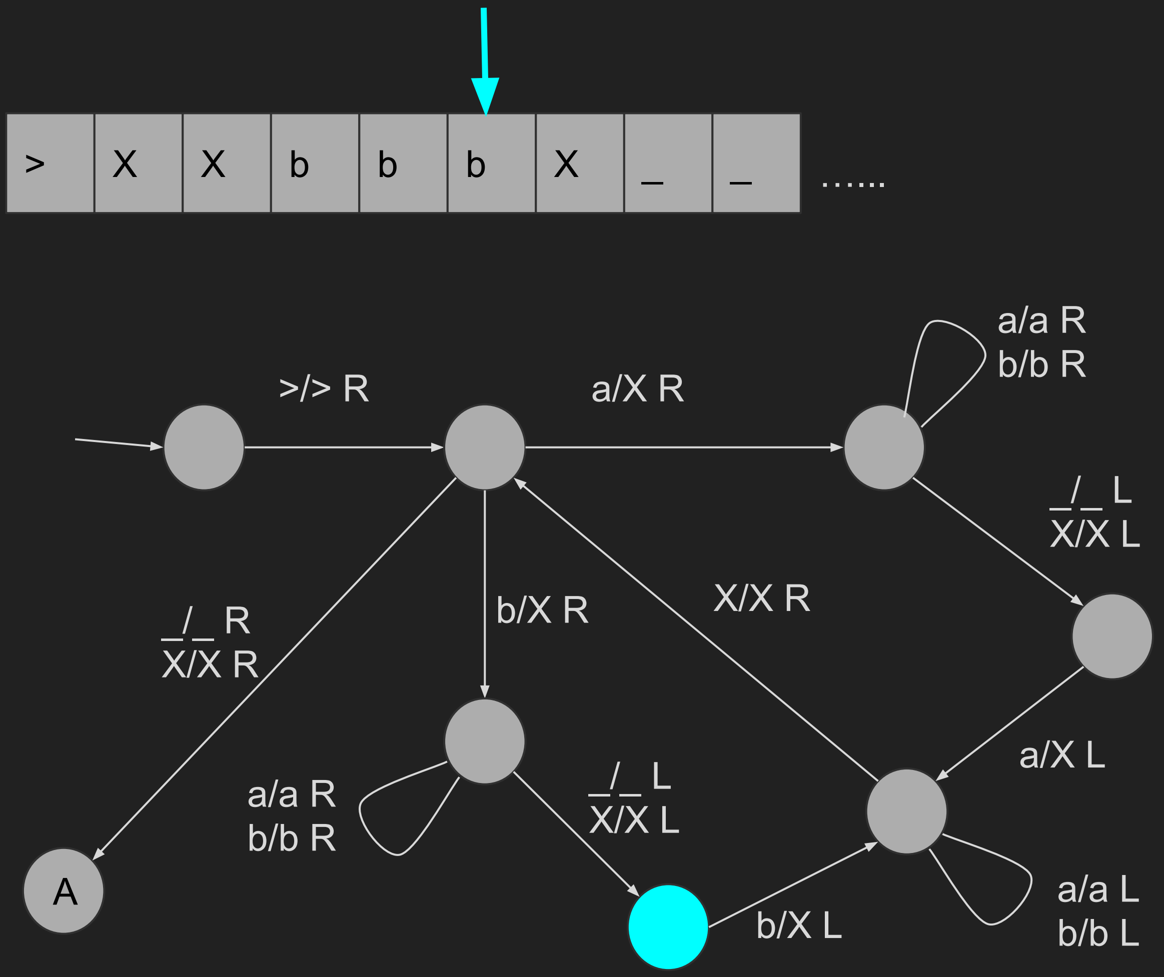
>
X
X
b
b
b
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
a/a L
b/b L
b/X L
A



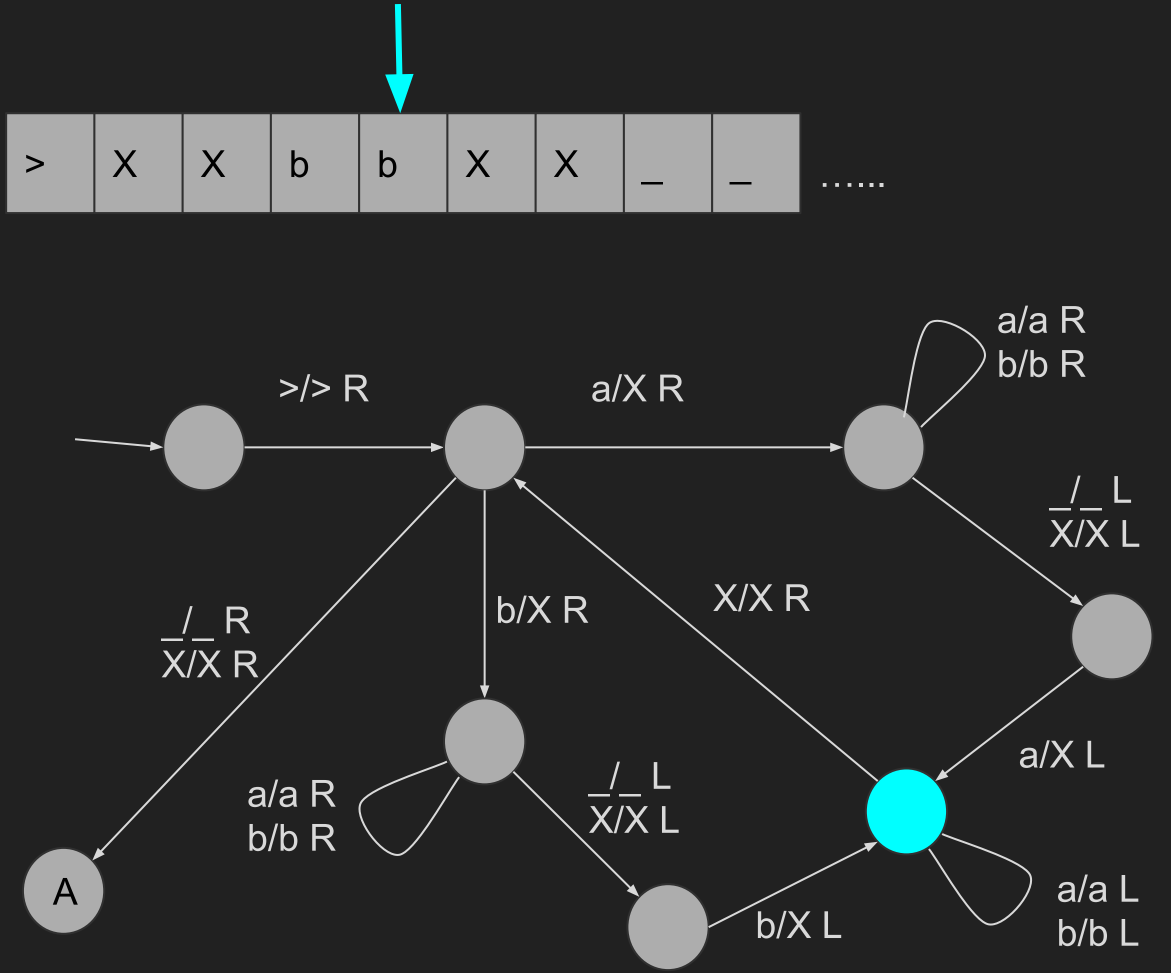
>
X
X
b
b
b
X
_
_
……
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



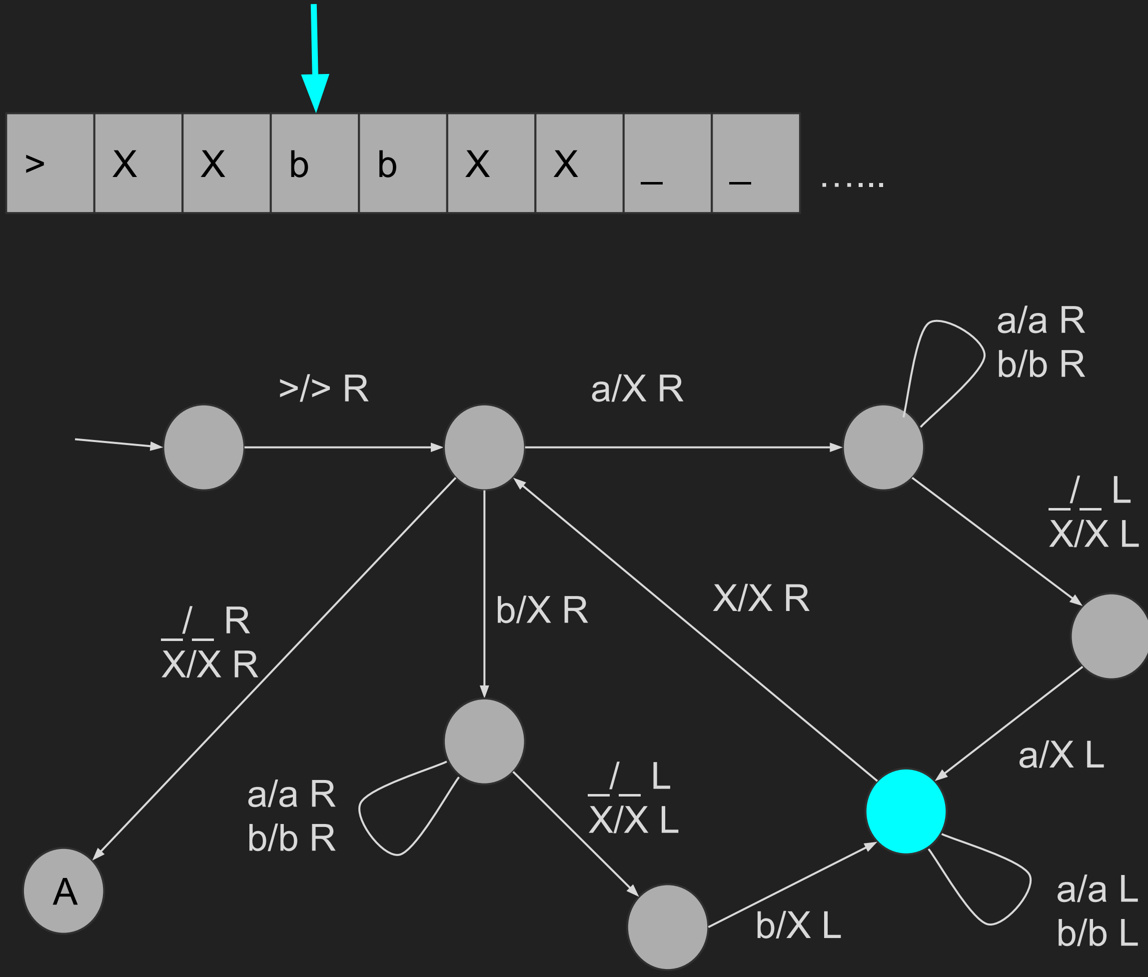
>
X
X
b
b
b
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
b/X R
X/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



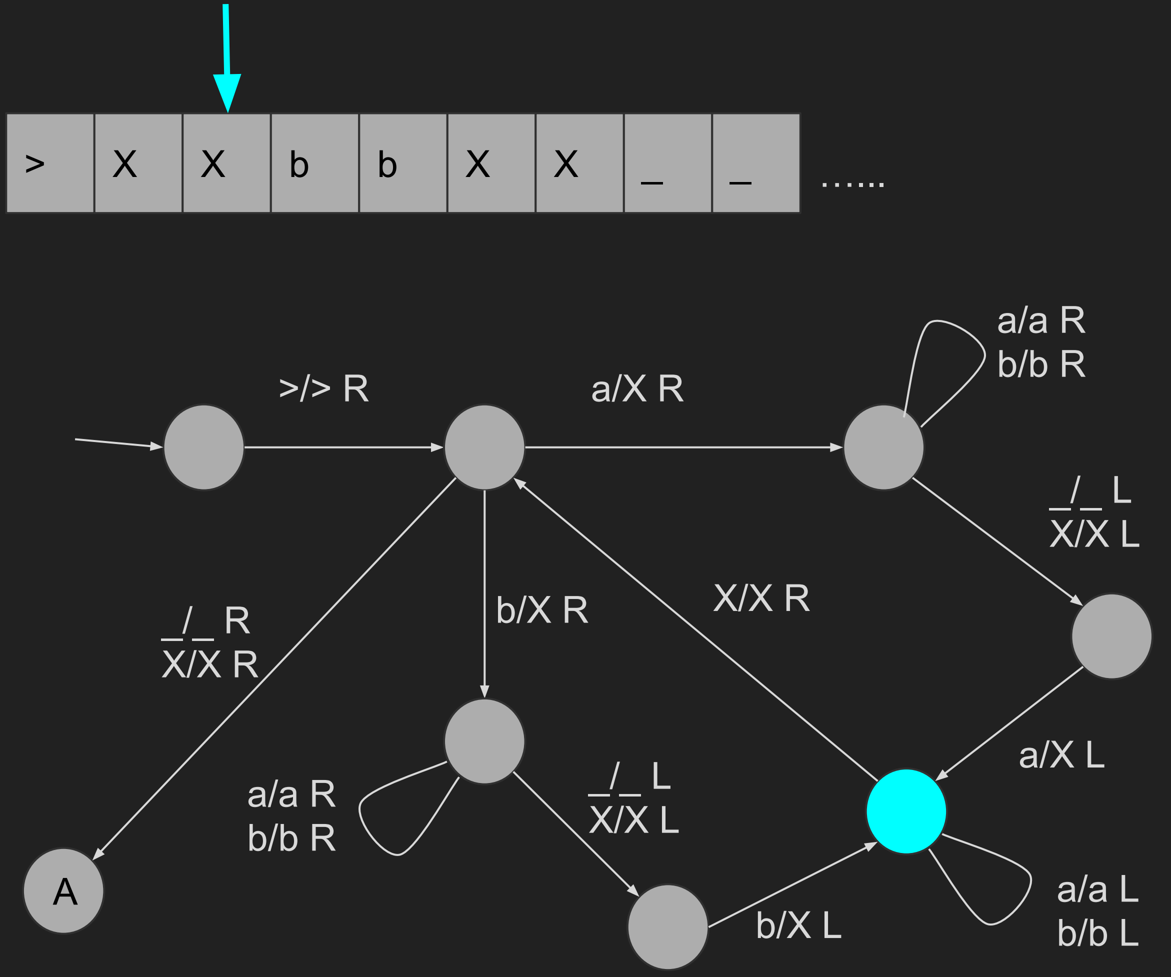
> X X b b b X _ _ ......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



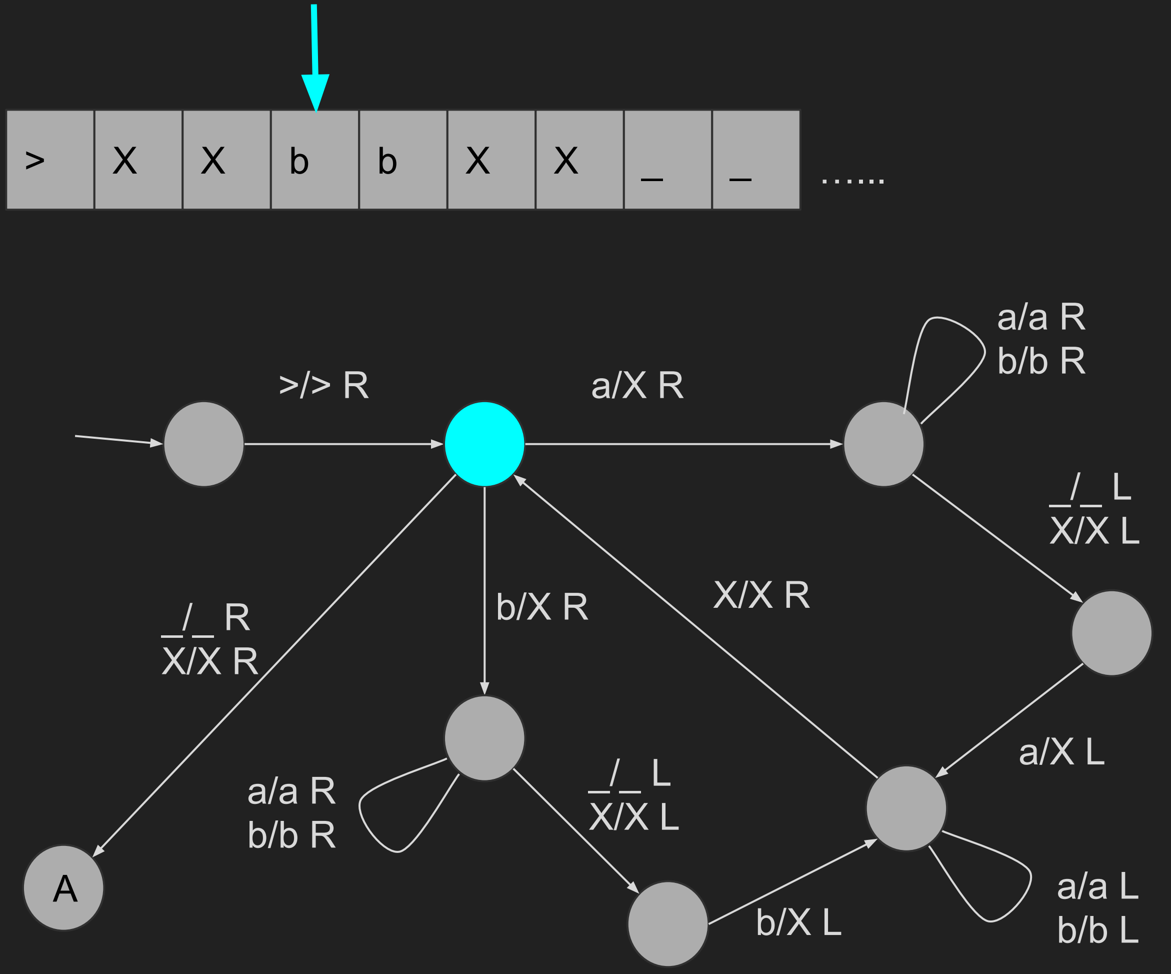
>
X
X
b
b
X
X
_
_
……
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



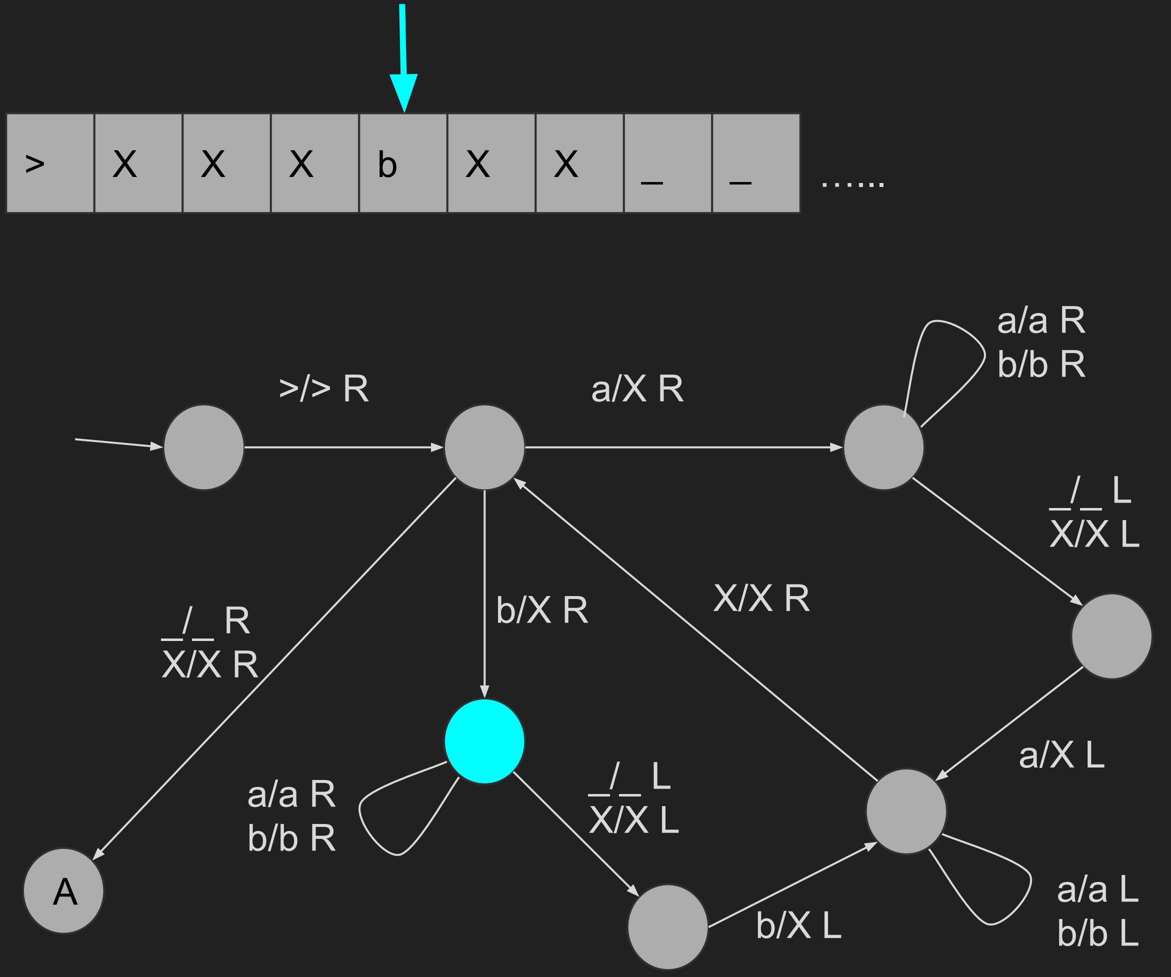
>
X
X
b
b
X
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L

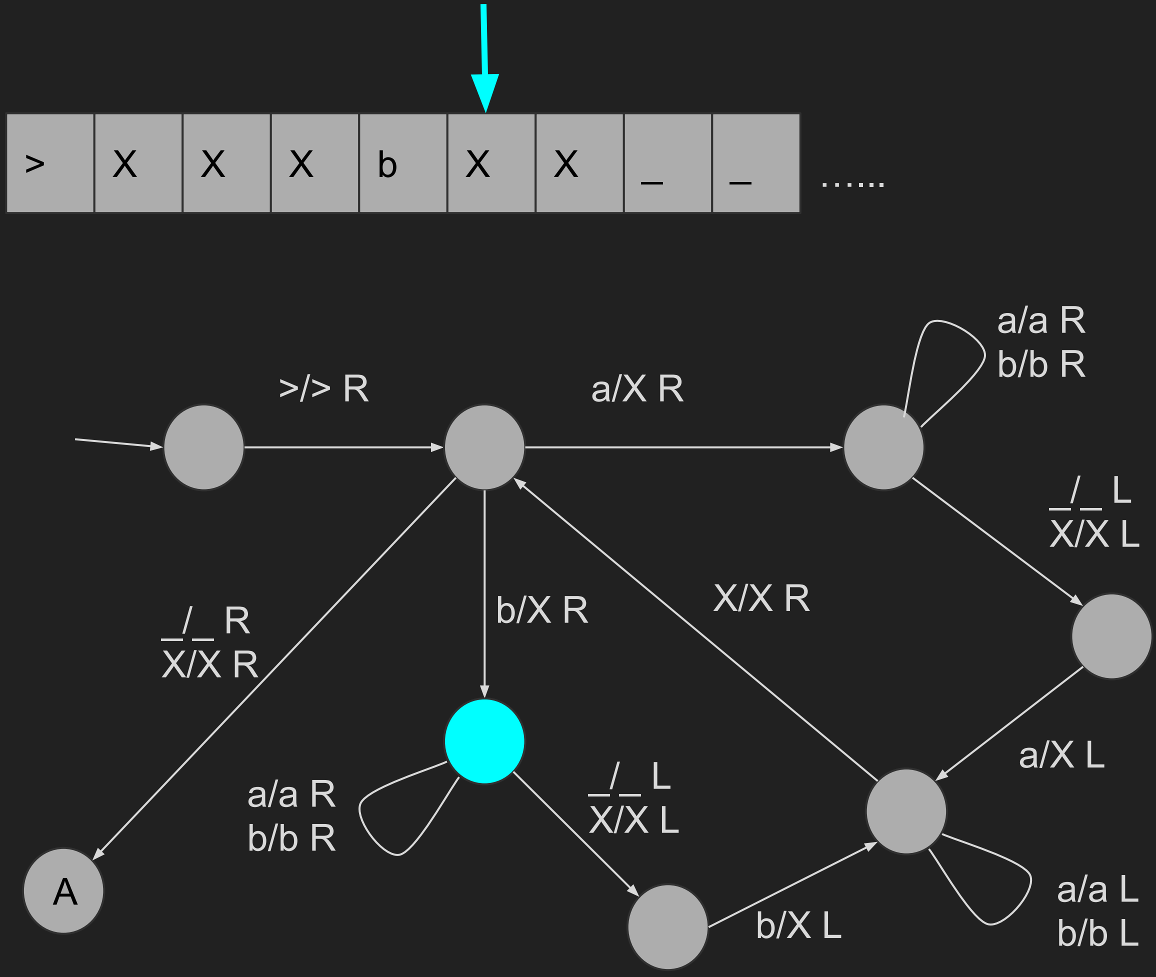


>
X
X
b
b
X
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
a/a L
b/b L
b/X L
A

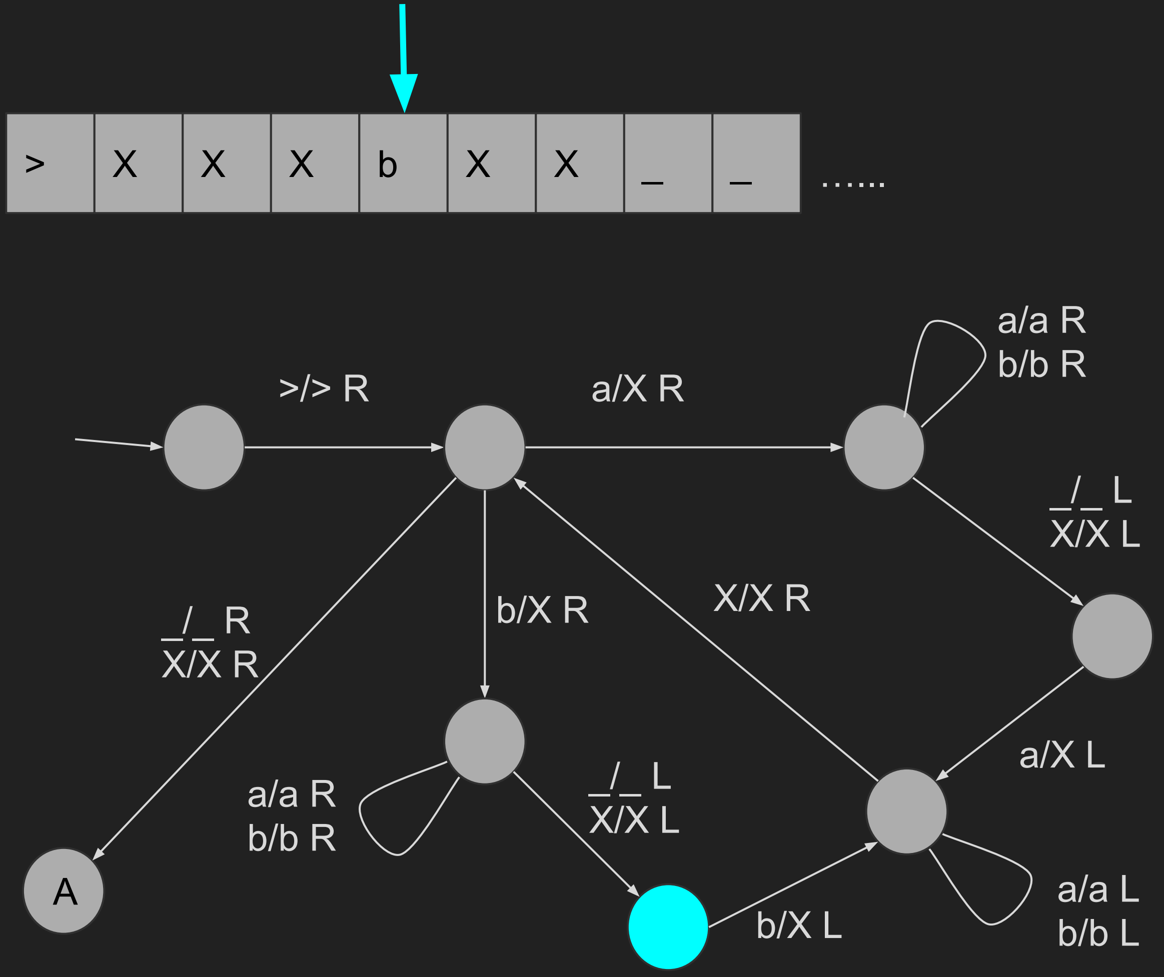


> X X b b X X _ _ ......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L

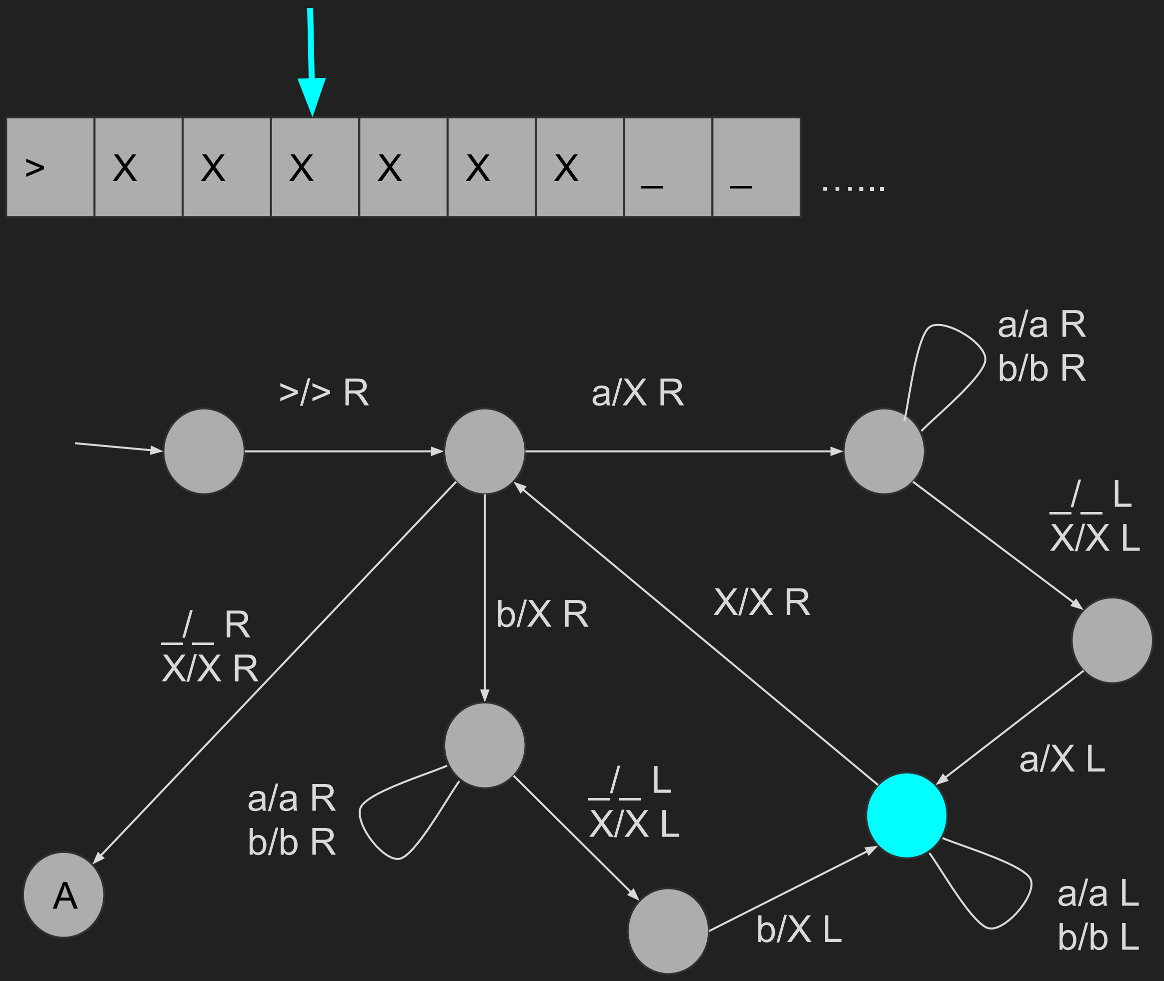

> X X X b X X _ _ ......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



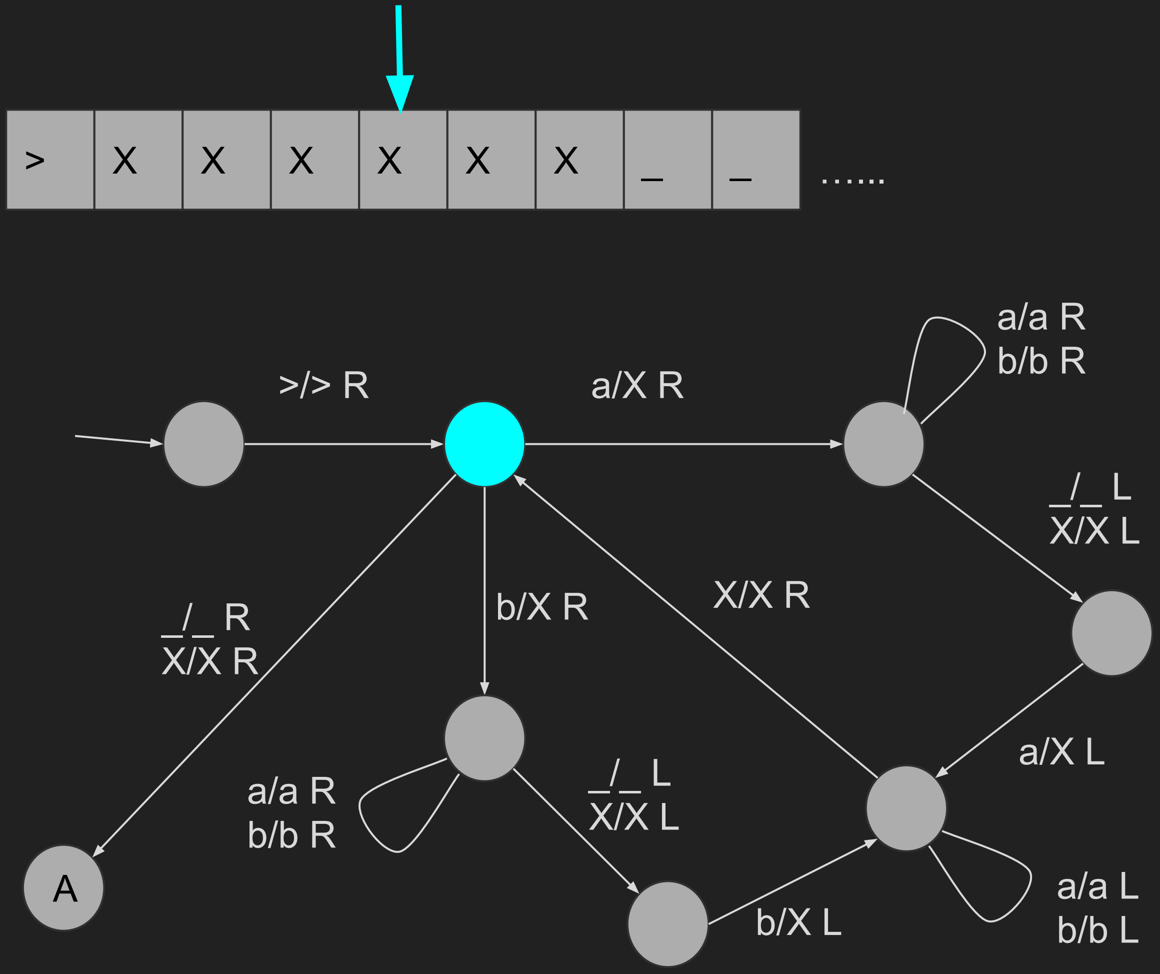
>
X
X
X
b
X
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



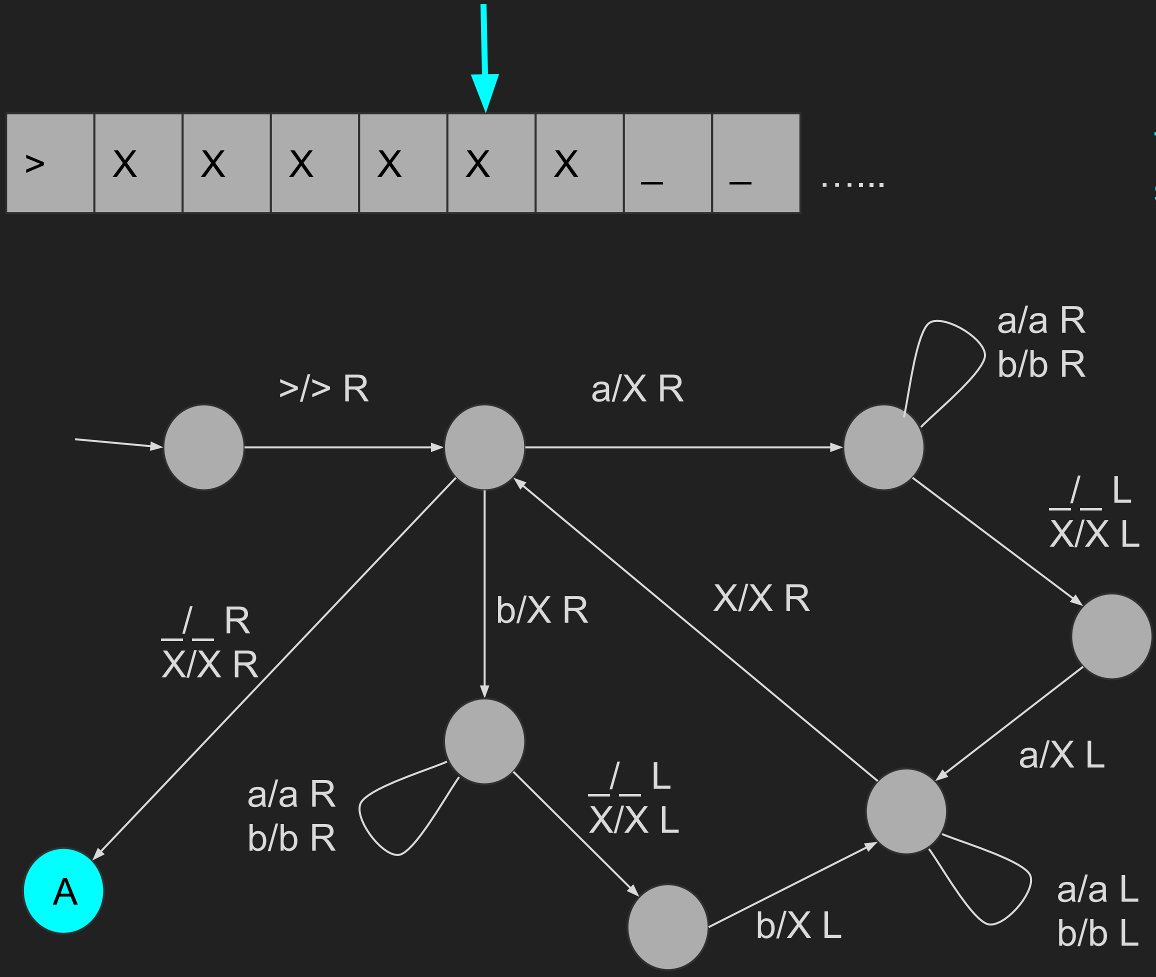
>
X
X
X
b
X
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



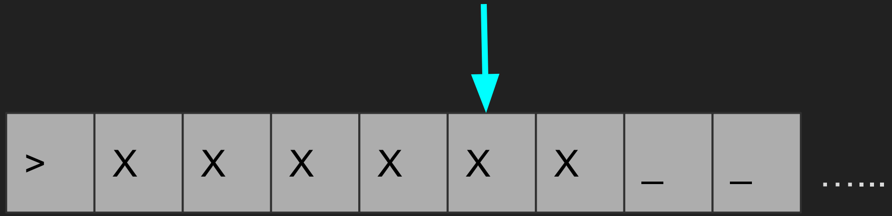
>
X
X
X
X
X
X
_
_
......
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
_/_ R
X/X R
b/X R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



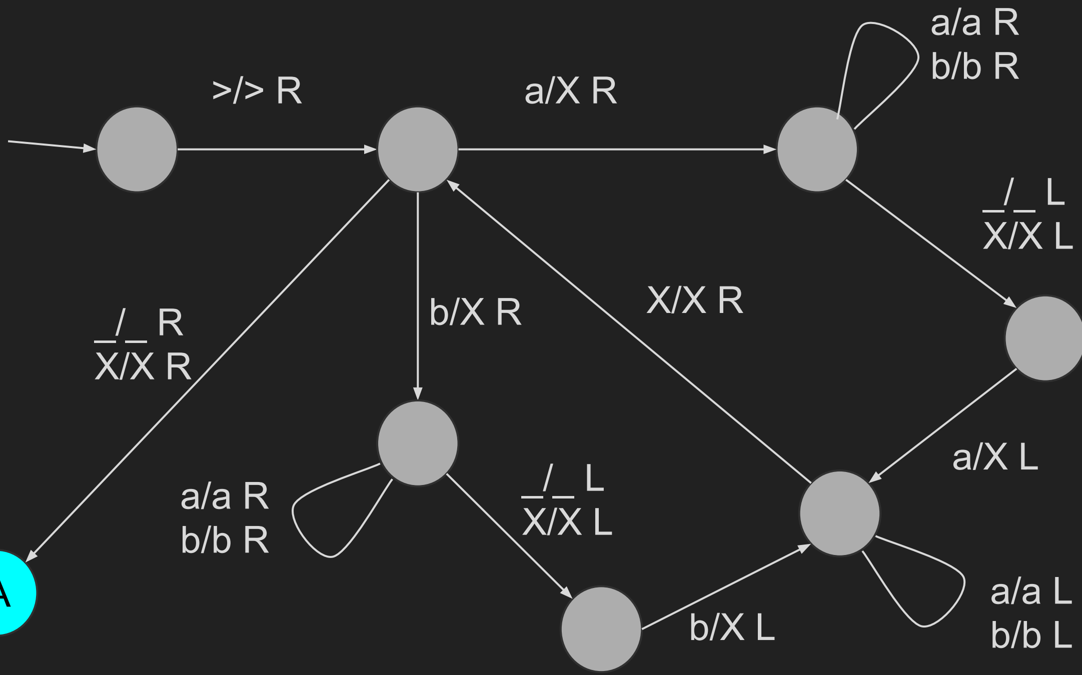
>
X
X
X
X
X
X
_
_
……
>/> R
a/X R
a/a R
b/b R
_/_ L
X/X L
X/X R
b/X R
_/_ R
X/X R
a/X L
a/a R
b/b R
_/_ L
X/X L
A
b/X L
a/a L
b/b L



The Turing machine accepts strings *abbbba*



The Turing machine accepts strings *abbbba*



This machine doesn't accept palindromes of odd length like *aabaa*

(1) what happens when you run the machine with input *aabaa*?

(2) can you fix the machine so that it accepts palindromes of odd and even length?

Formal definition

A deterministic Turing machine (TM) is a structure $M = (Q, \Gamma, \Sigma, \delta, q_s, q_{acc}, q_{rej})$ where

- Q is a finite set of states
- Γ is a finite tape alphabet
  - with two special symbols _ (blank) and > (leftmost marker)
- Σ is a finite input alphabet ($\Sigma \subseteq \Gamma$)
- δ is a transition function : $Q \times \Gamma \rightarrow Q \times \Gamma \times \{0,1\}$
  - $\delta(q, a) = (p, b, d)$ means that in state q with a in the cell under the pointer, go to state p after writing b in the cell and moving the pointer left ($d=0$) or right ($d=1$)
- q_s is the start state ($q_s \in Q$)
- q_{acc} and q_{rej} are the accept and reject states ($q_{acc}, q_{rej} \in Q$)

Turing machine execution

A finite state machine goes from state to state when it executes

A Turing machine goes from configuration to configuration when it executes

- Need to account for the content of the tape

A configuration (q, u, i) is a snapshot of the Turing machine during execution:

- q is the current state ($q \in Q$)
- u is the content of the tape ($u \in \Gamma^*$)
- i is the position of the pointer (i=0 is leftmost)

Turing machine execution

A Turing machine M starts in an input configuration (q_s, >w, 0)

- q_s is the start state of M
- >w is the leftmost marker followed by the input string w in Σ^*
- the pointer is initially pointing position 0 (leftmost cell)

M stops when it reaches a configuration of either form:

- (q_{acc}, u, i) for any u, i (an accepting configuration)
- (q_{rej}, u, i) for any u,i (a rejecting configuration)

Turing machine execution

How does the machine go from configuration to configuration?

Execution step: $c \Rightarrow d$:

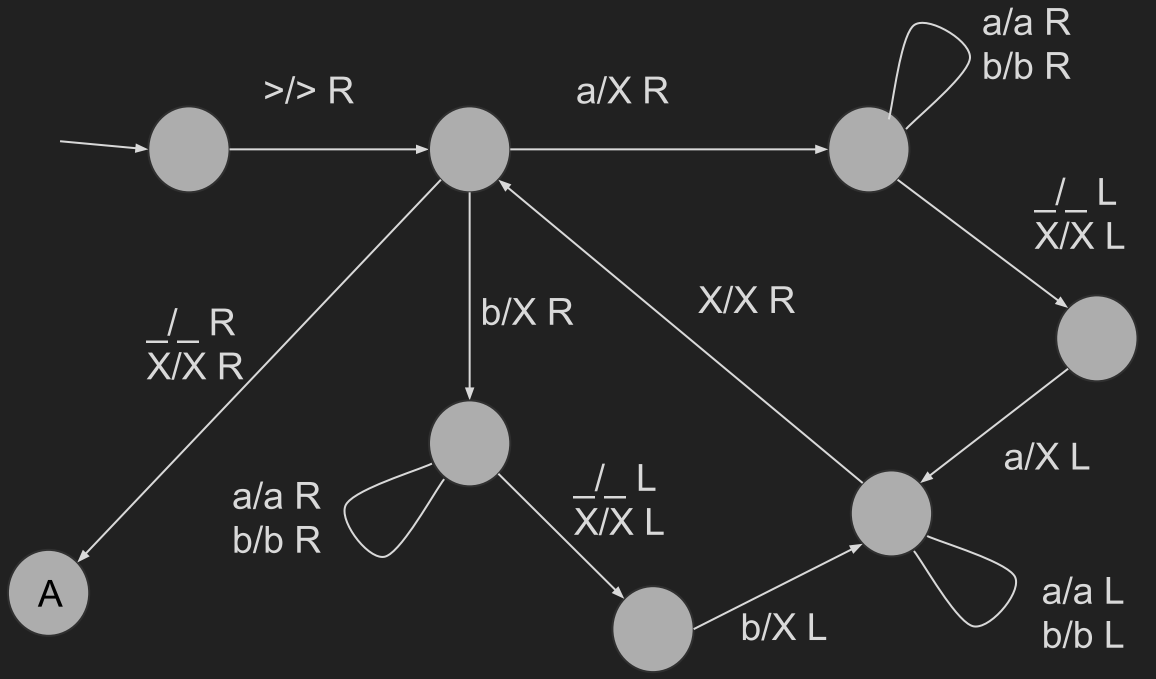
$(p, a_0 \ldots a_k, i) \Rightarrow (q, a_0 \ldots a_{i-1} b a_{i+1} \ldots a_k, i+1)$ if $\delta(p, a_i) = (q, b, 1)$

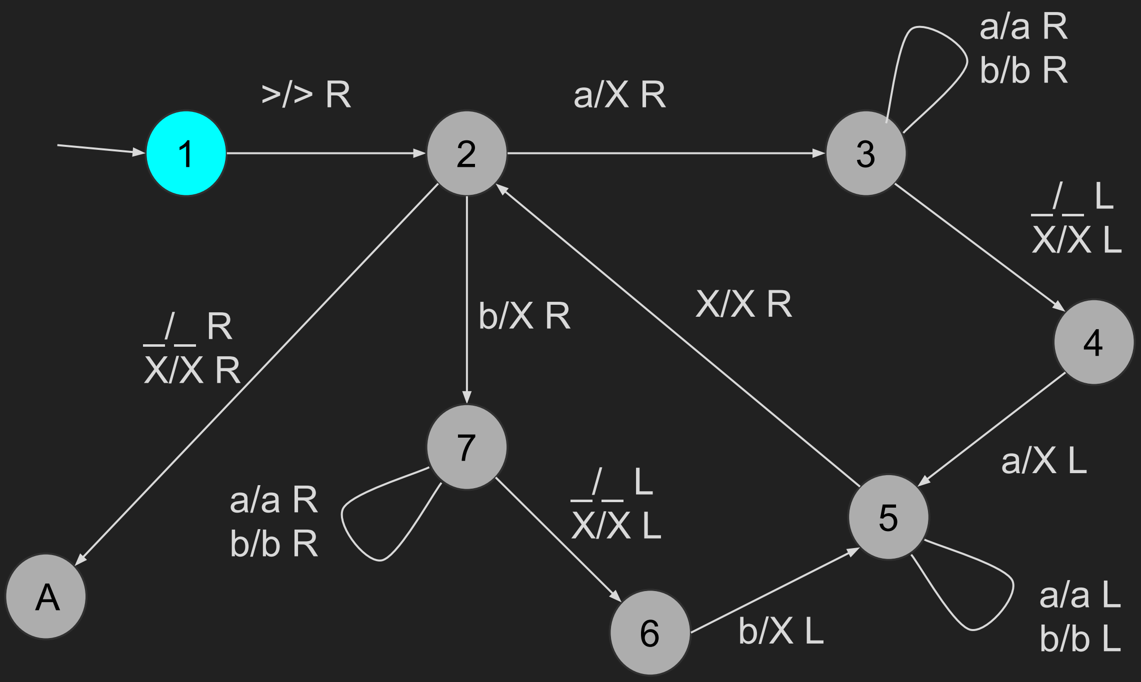
$(p, a_0 \ldots a_k, i) \Rightarrow (q, a_0 \ldots a_{i-1} b a_{i+1} \ldots a_k, i-1)$ if $\delta(p, a_i) = (q, b, 0)$

Execution multistep: $c \Rightarrow^* d$

$c \Rightarrow^* d$ if either $c = d$ OR $c \Rightarrow c_1 \Rightarrow \ldots \Rightarrow c_k \Rightarrow d$ for some $c_1, \ldots, c_k$ $(k \geq 0)$

Example — {u ∈ {a,b}* | u = rev(u), u even length}





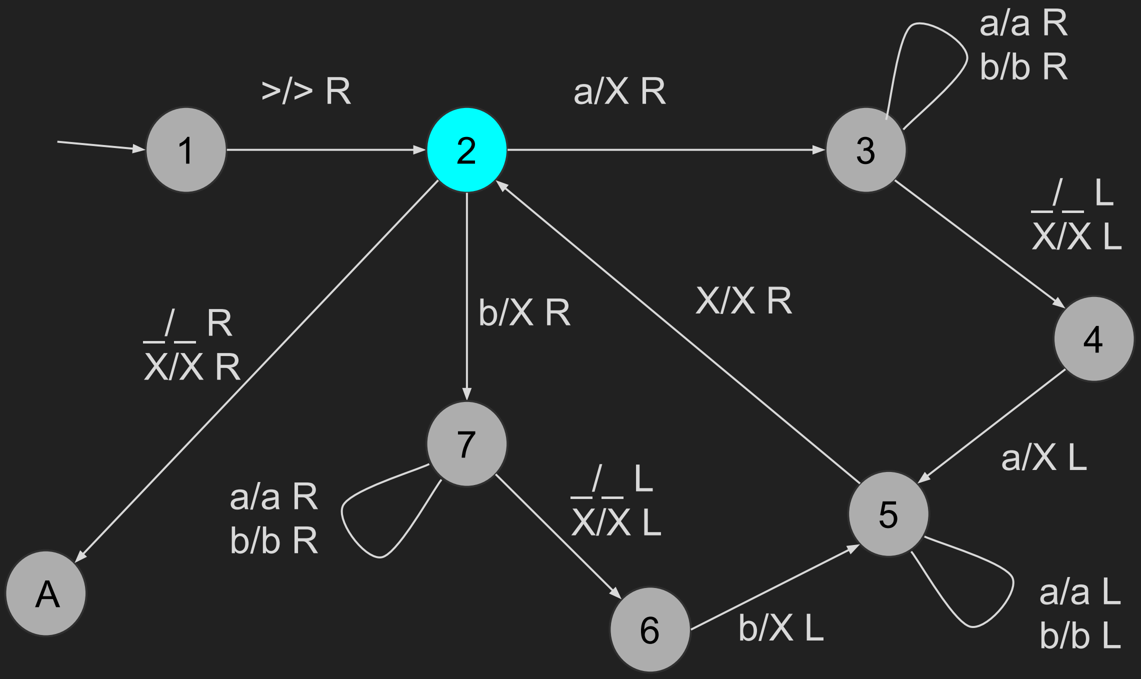
(1, >abbbba, 0)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



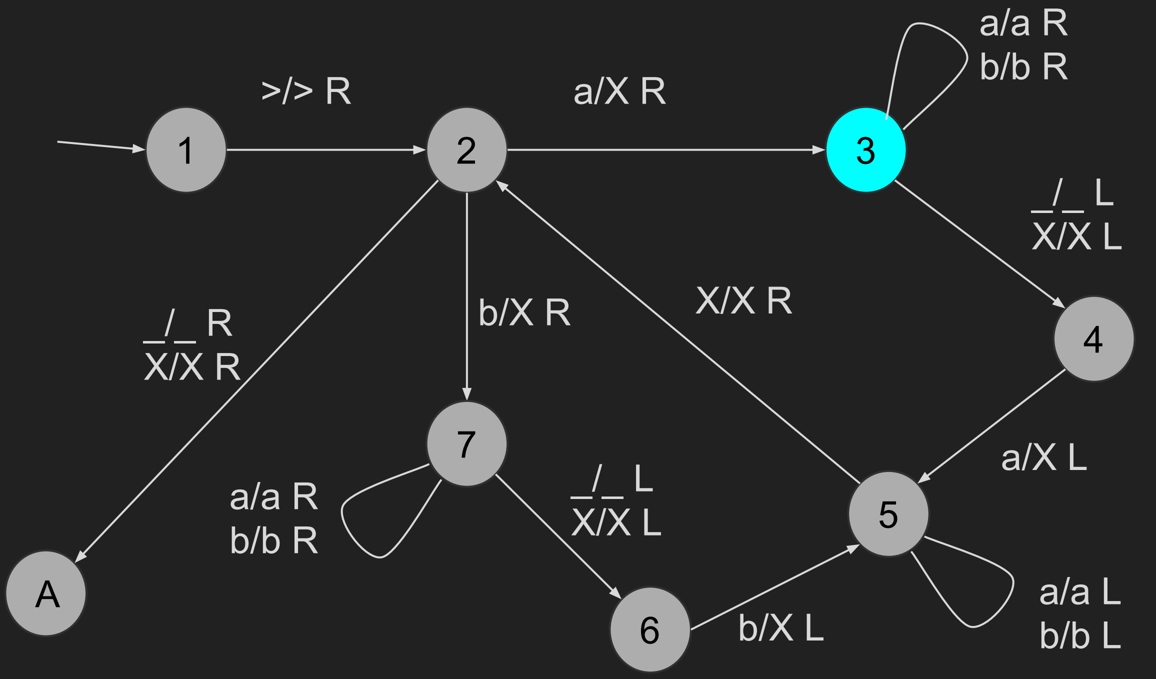
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)

$\partial(1,>) = (2,>,1)$
$\partial(2,\_) = (A,\_,1)$
$\partial(2,X) = (A,X,1)$
$\partial(2,a) = (3,X,1)$
$\partial(2,b) = (7,X,1)$
$\partial(3,a) = (3,a,1)$
$\partial(3,b) = (3,b,1)$

$\partial(3,\_) = (4,\_,0)$
$\partial(3,X) = (4,X,0)$
$\partial(4,a) = (5,X,0)$
$\partial(5,a) = (5,a,0)$
$\partial(5,b) = (5,b,0)$
$\partial(5,X) = (2,X,1)$
$\partial(6,b) = (5,X,0)$

$\partial(7,a) = (7,a,1)$
$\partial(7,b) = (7,b,1)$
$\partial(7,\_) = (6,\_,0)$
$\partial(7,X) = (6,X,0)$

For any other:
$\partial(q,s) = (R,s,1)$



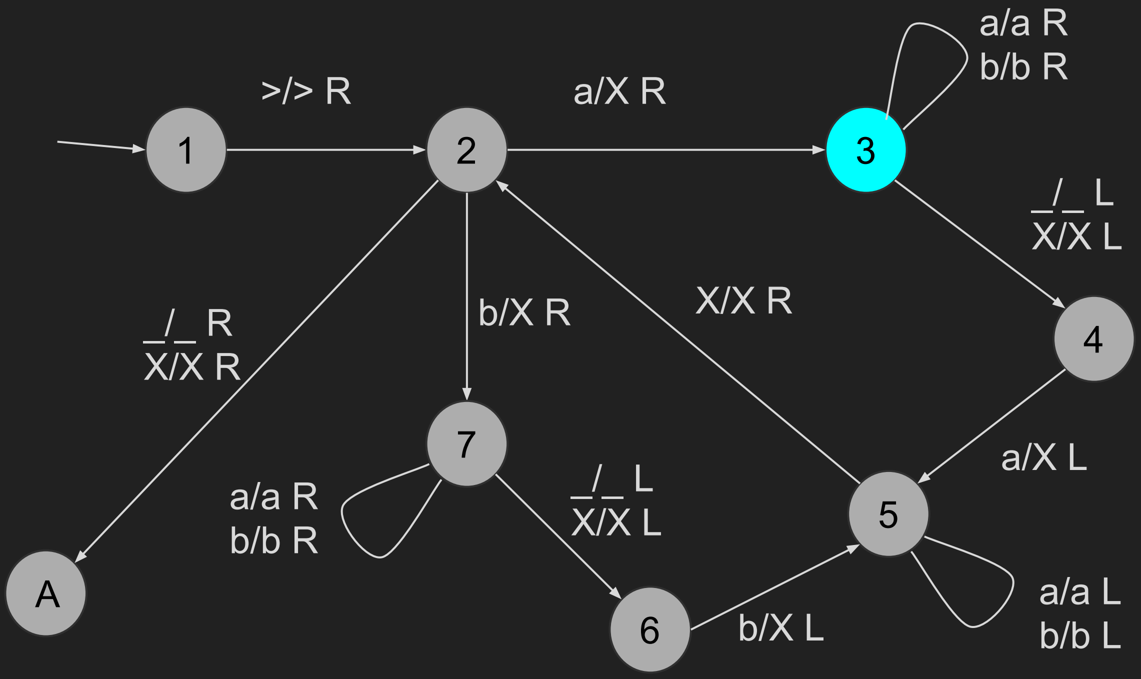
$(1, >abbbba, 0)$
$\Rightarrow (2, >abbbba, 1)$
$\Rightarrow (3, >Xbbbba, 2)$

$\partial(1,>) = (2,>,1)$
$\partial(2,\_) = (A,\_,1)$
$\partial(2,X) = (A,X,1)$
$\partial(2,a) = (3,X,1)$
$\partial(2,b) = (7,X,1)$
$\partial(3,a) = (3,a,1)$
$\partial(3,b) = (3,b,1)$

$\partial(3,\_) = (4,\_,0)$
$\partial(3,X) = (4,X,0)$
$\partial(4,a) = (5,X,0)$
$\partial(5,a) = (5,a,0)$
$\partial(5,b) = (5,b,0)$
$\partial(5,X) = (2,X,1)$
$\partial(6,b) = (5,X,0)$

$\partial(7,a) = (7,a,1)$
$\partial(7,b) = (7,b,1)$
$\partial(7,\_) = (6,\_,0)$
$\partial(7,X) = (6,X,0)$

For any other:
$\partial(q,s) = (R,s,1)$



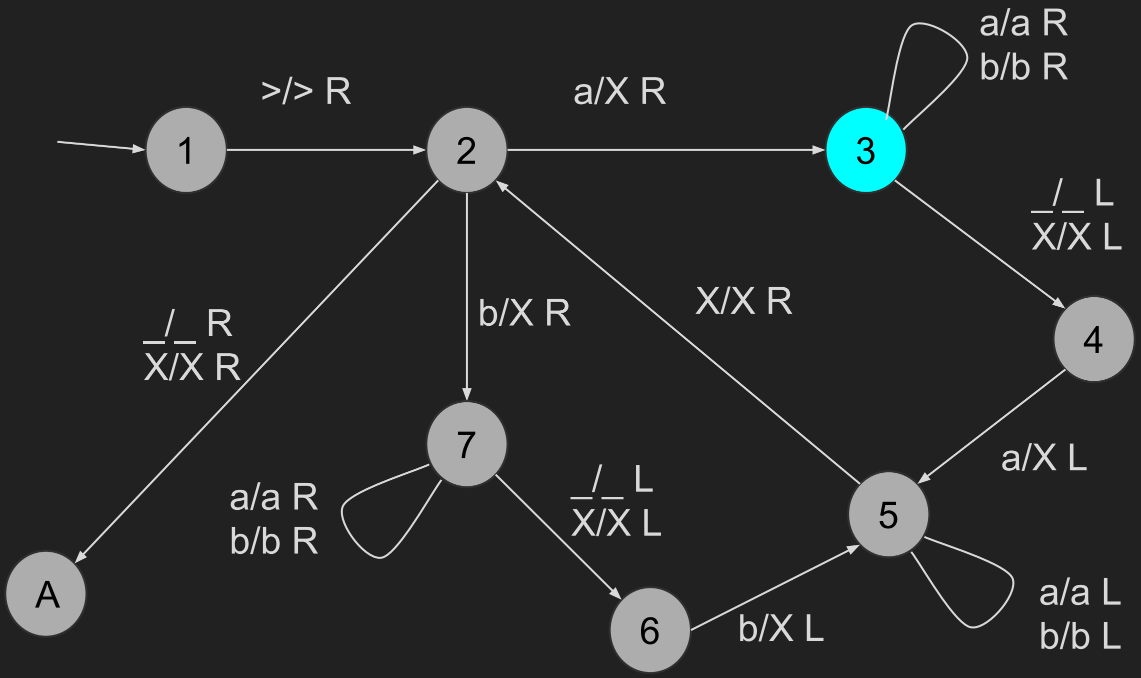
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



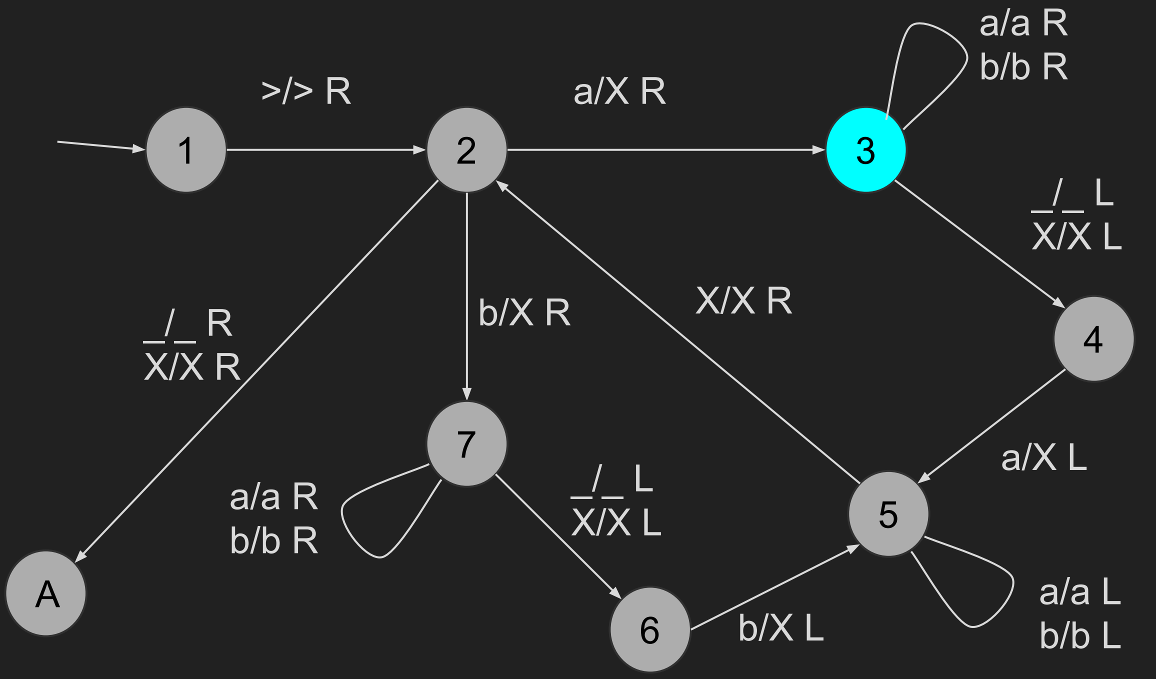
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



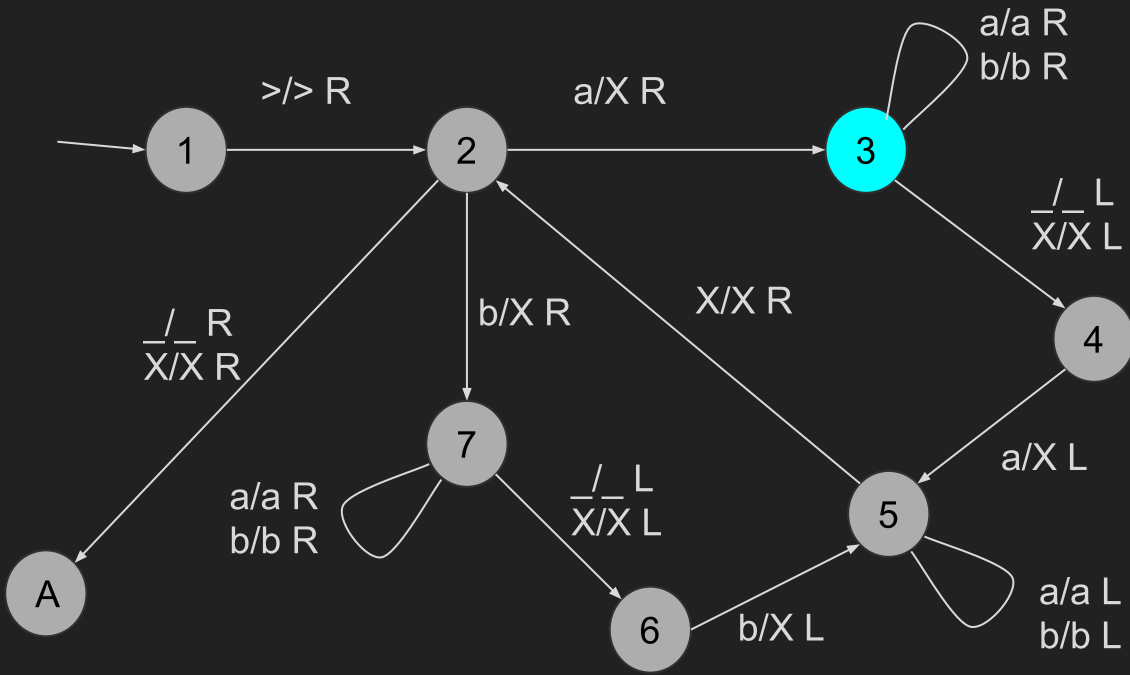
$$
\begin{aligned}
& (1, >abbbba, 0) \\
\Rightarrow\ & (2, >abbbba, 1) \\
\Rightarrow\ & (3, >Xbbbba, 2) \\
\Rightarrow\ & (3, >Xbbbba, 3) \\
\Rightarrow\ & (3, >Xbbbba, 4) \\
\Rightarrow\ & (3, >Xbbbba, 5)
\end{aligned}
$$

$\partial(1,>) = (2,>,1)$
$\partial(2,\_) = (A,\_,1)$
$\partial(2,X) = (A,X,1)$
$\partial(2,a) = (3,X,1)$
$\partial(2,b) = (7,X,1)$
$\partial(3,a) = (3,a,1)$
$\partial(3,b) = (3,b,1)$

$\partial(3,\_) = (4,\_,0)$
$\partial(3,X) = (4,X,0)$
$\partial(4,a) = (5,X,0)$
$\partial(5,a) = (5,a,0)$
$\partial(5,b) = (5,b,0)$
$\partial(5,X) = (2,X,1)$
$\partial(6,b) = (5,X,0)$

$\partial(7,a) = (7,a,1)$
$\partial(7,b) = (7,b,1)$
$\partial(7,\_) = (6,\_,0)$
$\partial(7,X) = (6,X,0)$

For any other:
$\partial(q,s) = (R,s,1)$



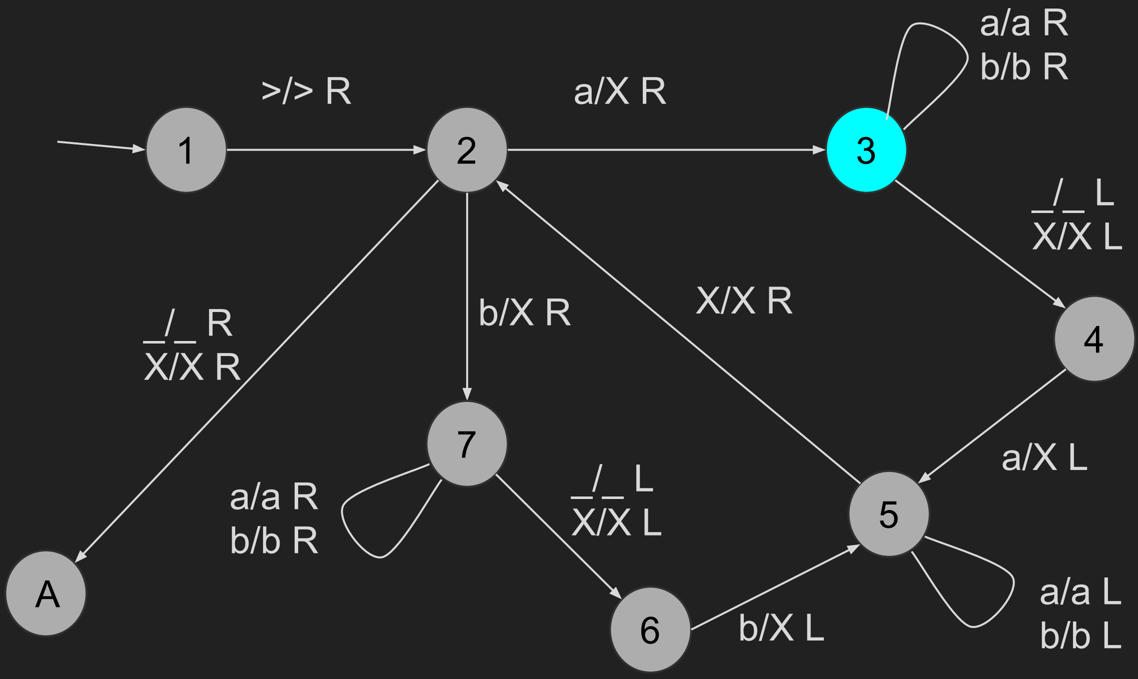
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



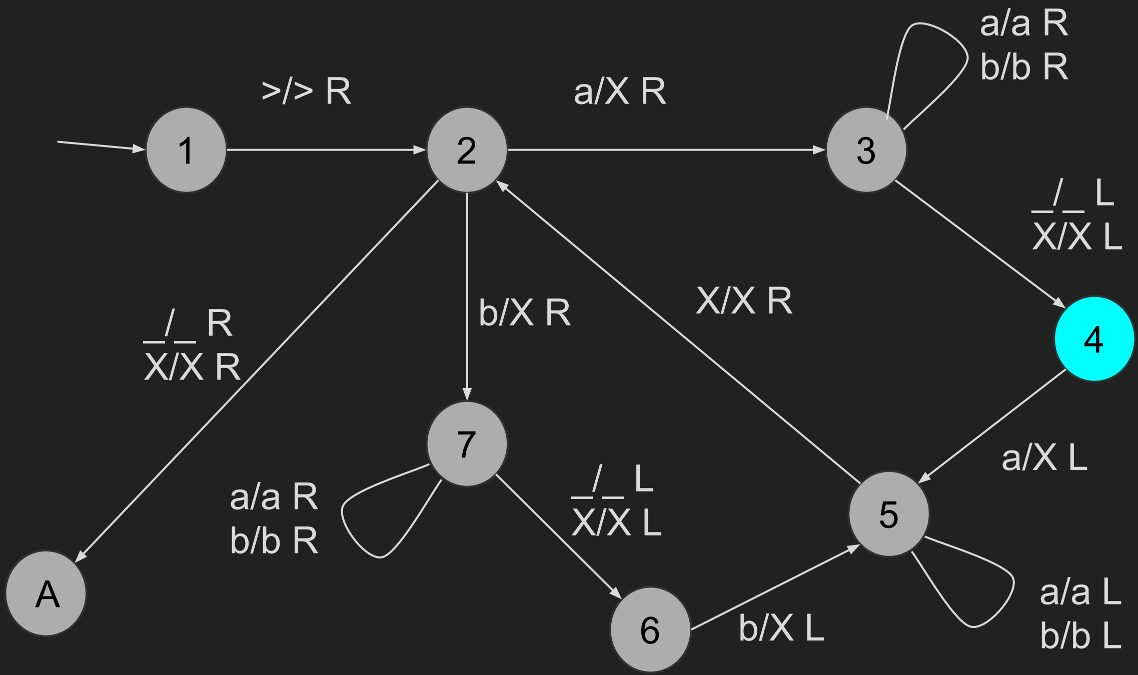
(1, >abbbba, 0)
$\Rightarrow$ (2, >abbbba, 1)
$\Rightarrow$ (3, >Xbbbba, 2)
$\Rightarrow$ (3, >Xbbbba, 3)
$\Rightarrow$ (3, >Xbbbba, 4)
$\Rightarrow$ (3, >Xbbbba, 5)
$\Rightarrow$ (3, >Xbbbba, 6)
$\Rightarrow$ (3, >Xbbbba_, 7)

$\partial(1,>) = (2,>,1)$
$\partial(2,\_) = (A,\_,1)$
$\partial(2,X) = (A,X,1)$
$\partial(2,a) = (3,X,1)$
$\partial(2,b) = (7,X,1)$
$\partial(3,a) = (3,a,1)$
$\partial(3,b) = (3,b,1)$

$\partial(3,\_) = (4,\_,0)$
$\partial(3,X) = (4,X,0)$
$\partial(4,a) = (5,X,0)$
$\partial(5,a) = (5,a,0)$
$\partial(5,b) = (5,b,0)$
$\partial(5,X) = (2,X,1)$
$\partial(6,b) = (5,X,0)$

$\partial(7,a) = (7,a,1)$
$\partial(7,b) = (7,b,1)$
$\partial(7,\_) = (6,\_,0)$
$\partial(7,X) = (6,X,0)$

For any other:
$\partial(q,s) = (R,s,1)$



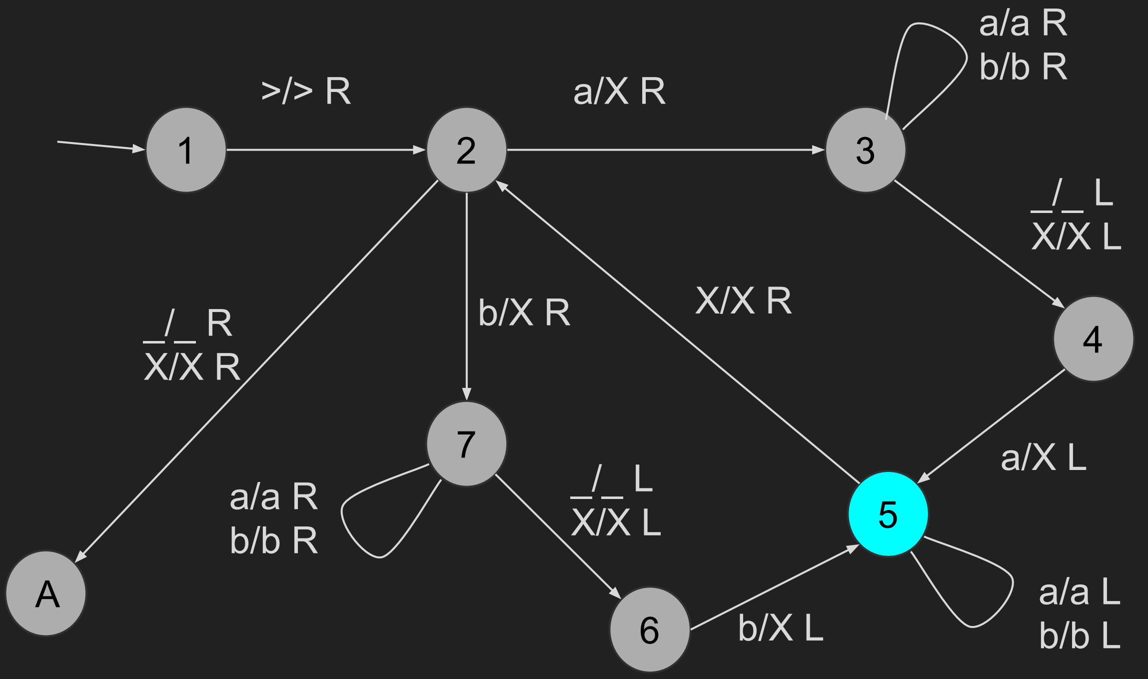
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



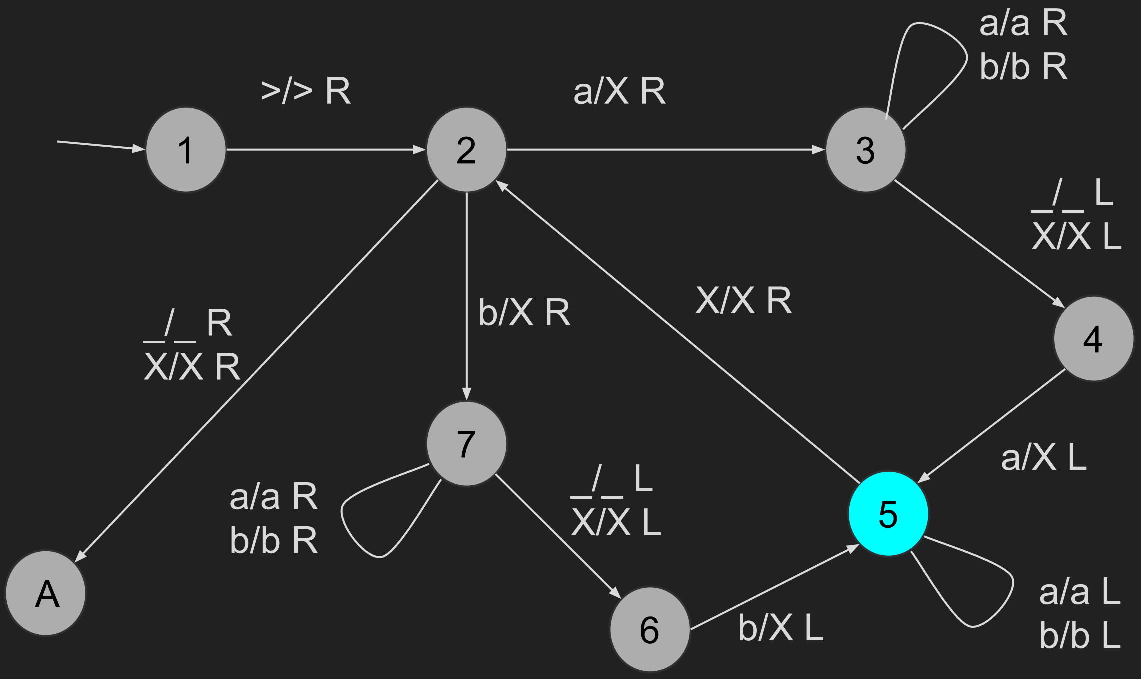
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



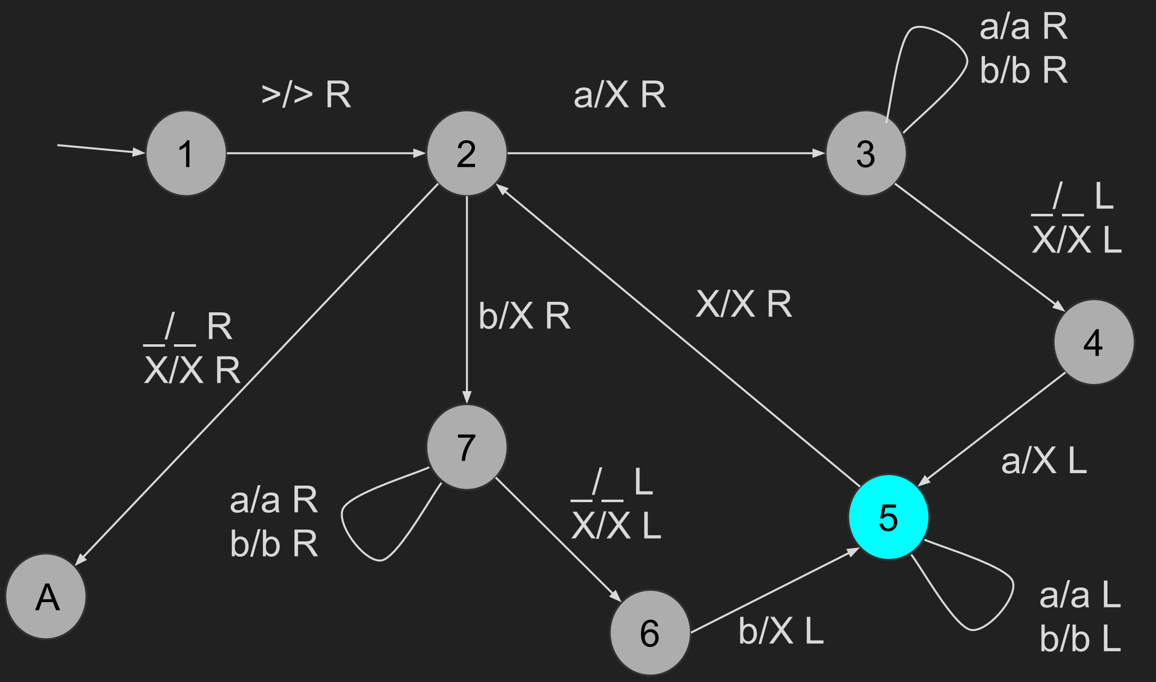
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



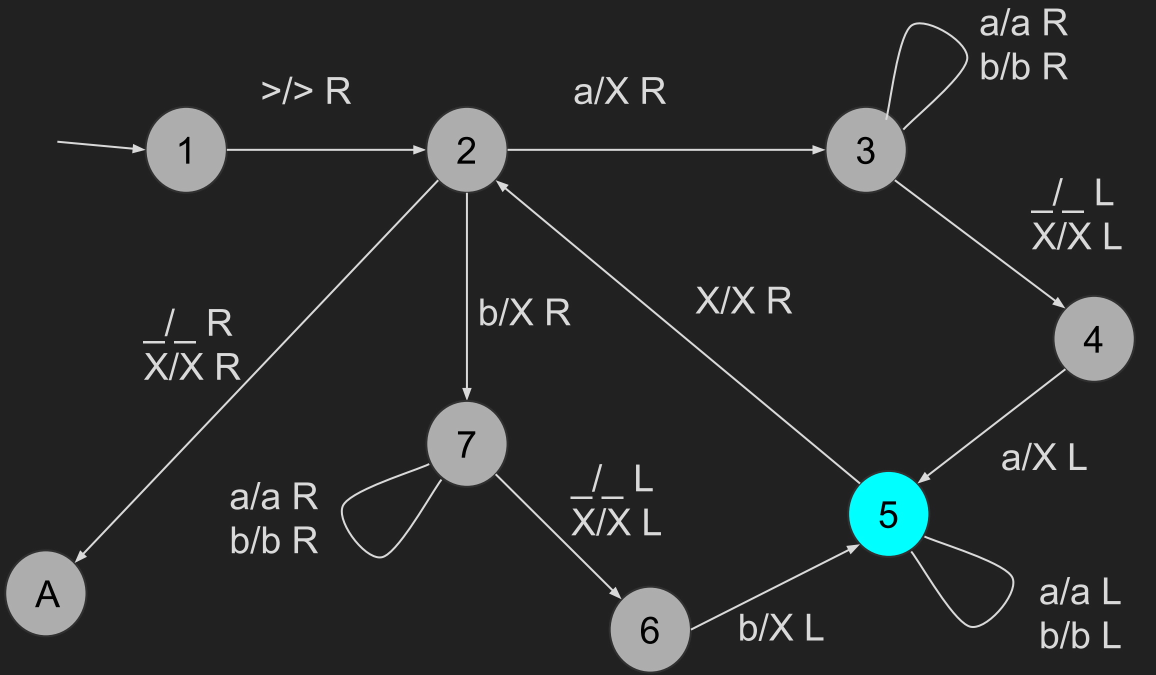
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



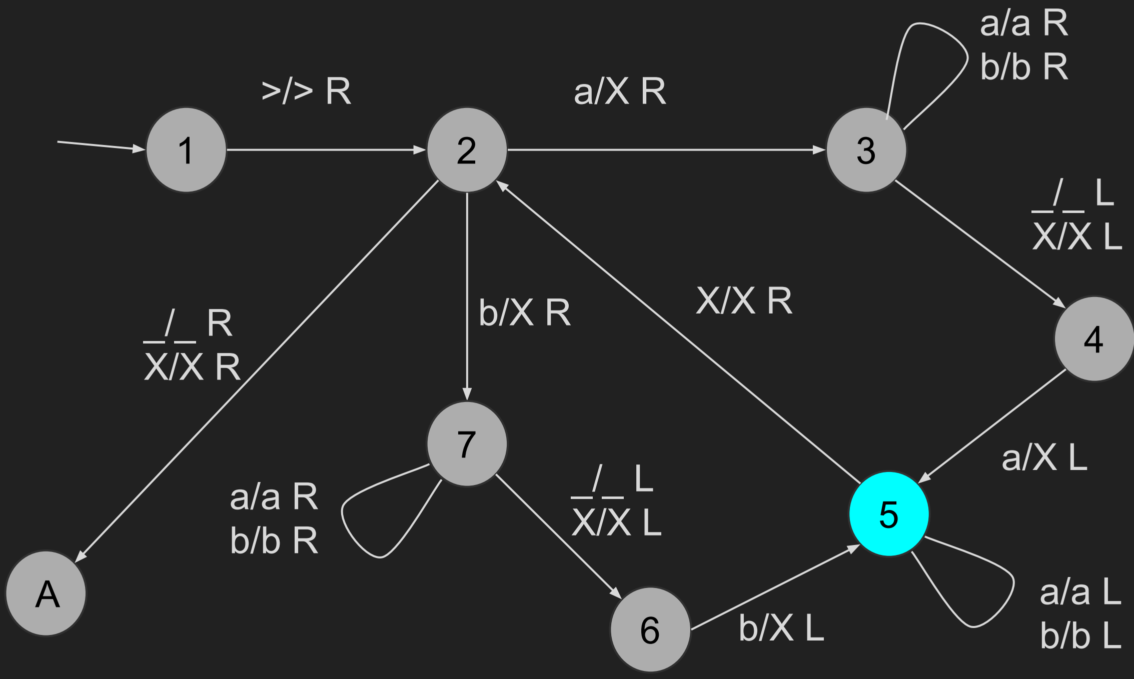
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)
⇒ (5, >XbbbbX_, 2)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



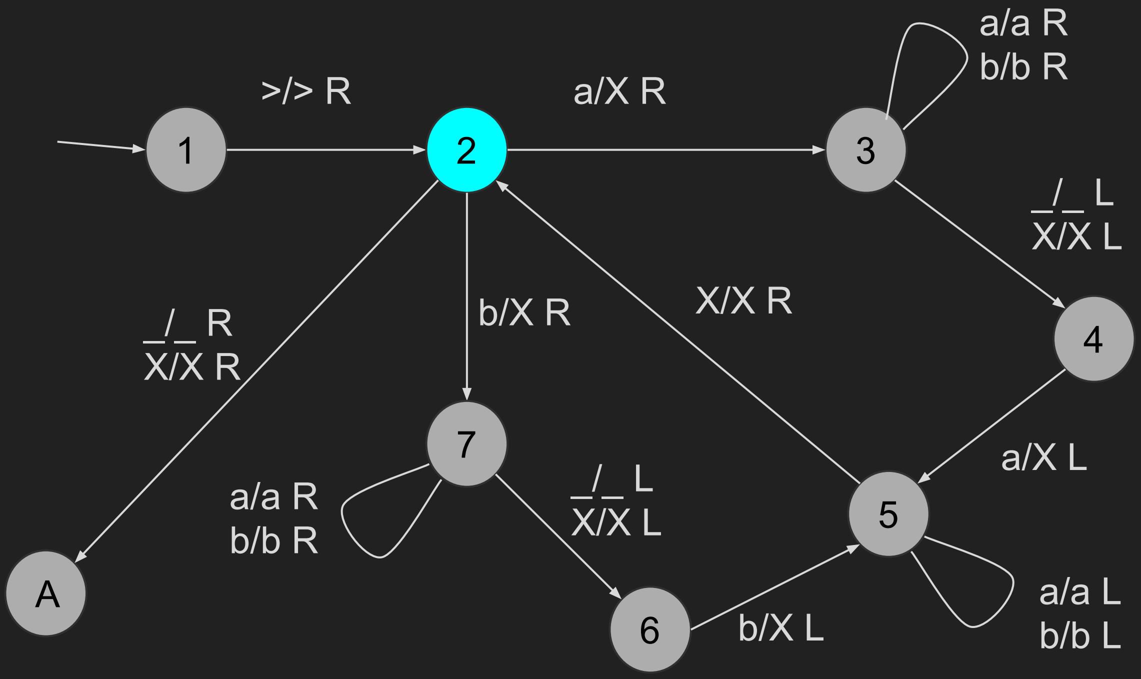
(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)
⇒ (5, >XbbbbX_, 2)
⇒ (5, >XbbbbX_, 1)

∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)



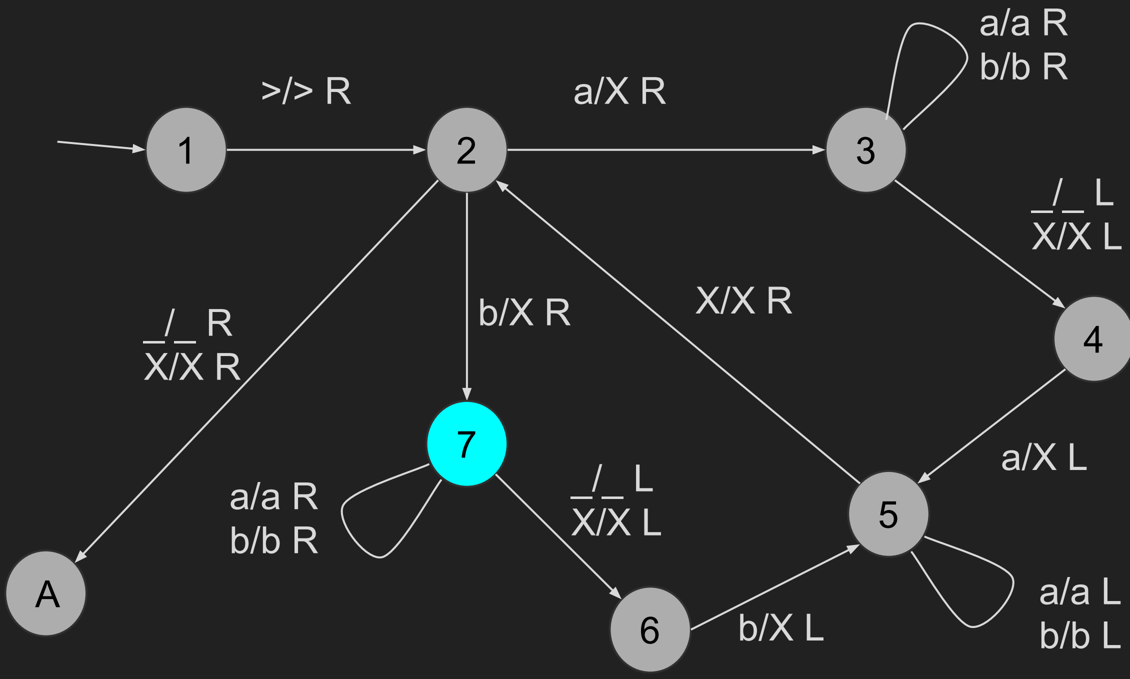
∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

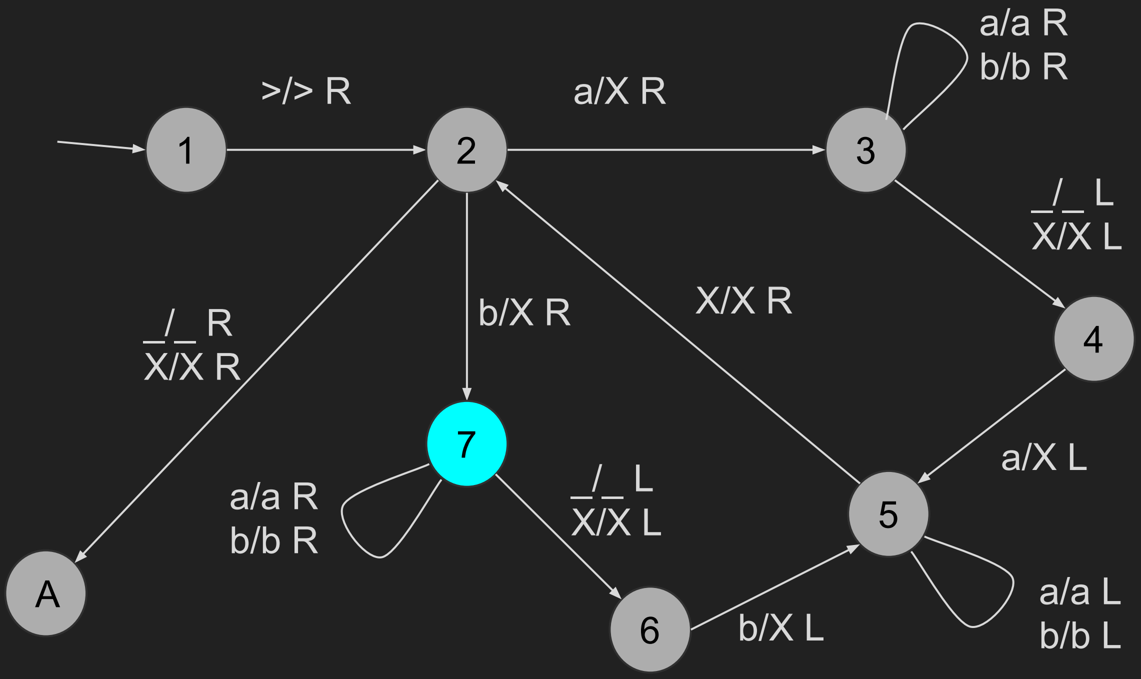
For any other:
∂(q,s) = (R,s,1)

(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)
⇒ (5, >XbbbbX_, 2)
⇒ (5, >XbbbbX_, 1)
⇒ (2, >XbbbbX_, 2)



∂(1,>) = (2,>,1)	∂(3,_) = (4,_,0)	∂(7,a) = (7,a,1)
∂(2,_) = (A,_,1)	∂(3,X) = (4,X,0)	∂(7,b) = (7,b,1)
∂(2,X) = (A,X,1)	∂(4,a) = (5,X,0)	∂(7,_) = (6,_,0)
∂(2,a) = (3,X,1)	∂(5,a) = (5,a,0)	∂(7,X) = (6,X,0)
∂(2,b) = (7,X,1)	∂(5,b) = (5,b,0)	
∂(3,a) = (3,a,1)	∂(5,X) = (2,X,1)	*For any other:*
∂(3,b) = (3,b,1)	∂(6,b) = (5,X,0)	*∂(q,s) = (R,s,1)*

(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)
⇒ (5, >XbbbbX_, 2)
⇒ (5, >XbbbbX_, 1)
⇒ (2, >XbbbbX_, 2)
⇒ (7, >XXbbbX_, 3)



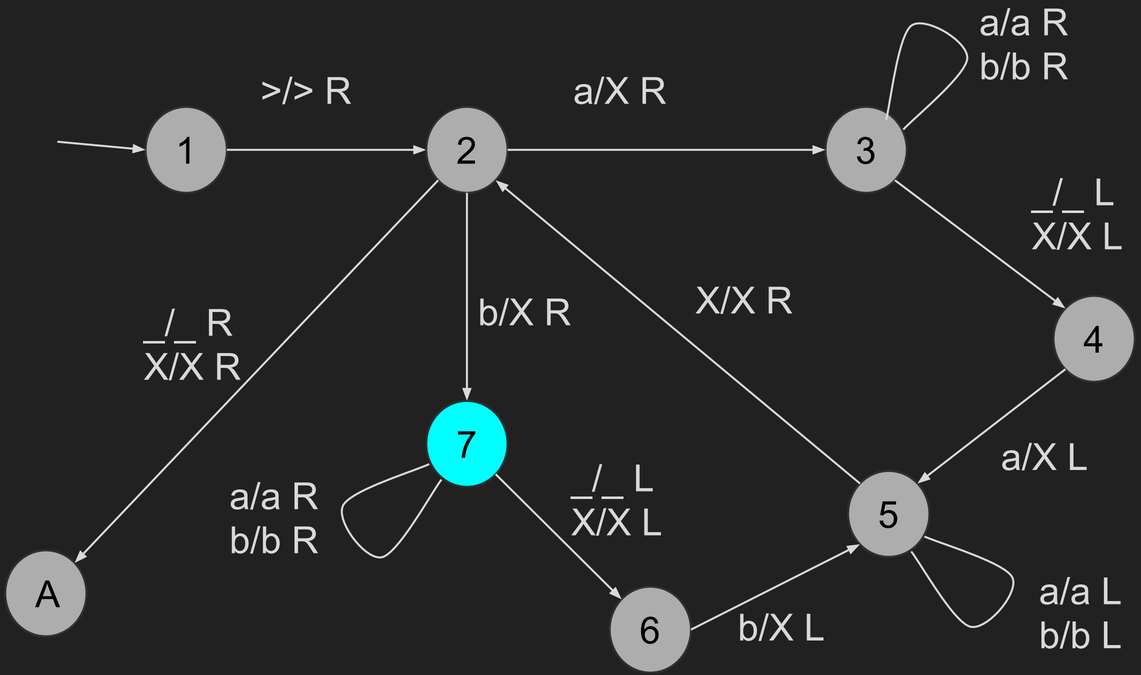
∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)

(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)
⇒ (5, >XbbbbX_, 2)
⇒ (5, >XbbbbX_, 1)
⇒ (2, >XbbbbX_, 2)
⇒ (7, >XXbbbX_, 3)
⇒ (7, >XXbbbX_, 4)



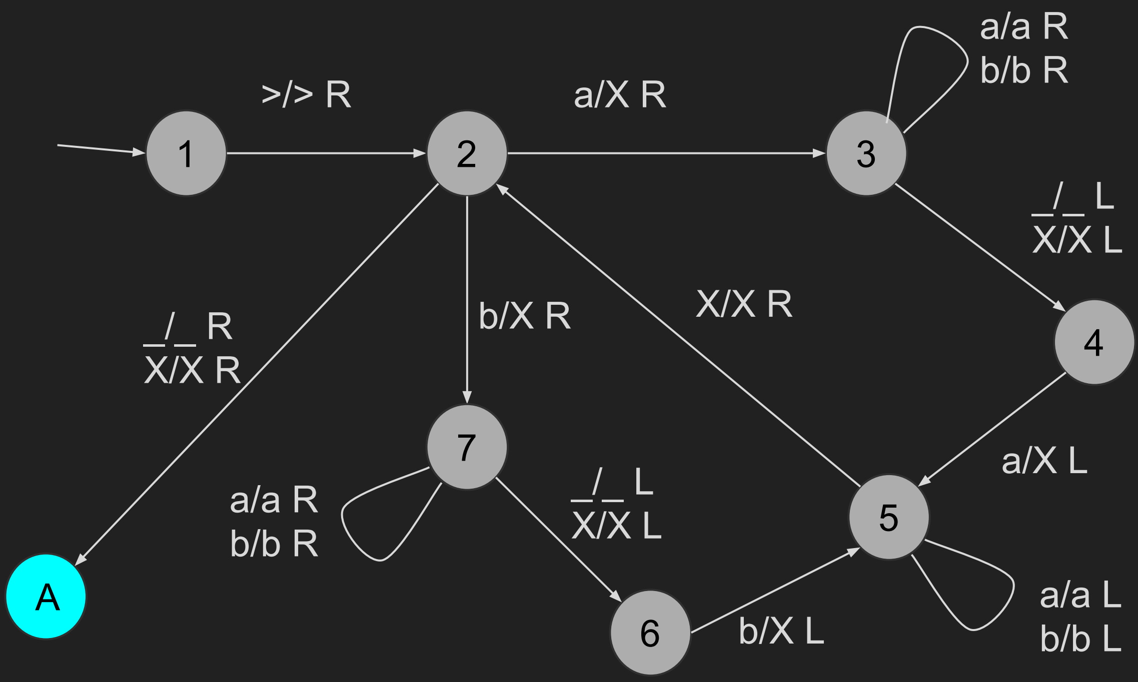
∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)

(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)
⇒ (5, >XbbbbX_, 2)
⇒ (5, >XbbbbX_, 1)
⇒ (2, >XbbbbX_, 2)
⇒ (7, >XXbbbX_, 3)
⇒ (7, >XXbbbX_, 4)
⇒ …



∂(1,>) = (2,>,1)
∂(2,_) = (A,_,1)
∂(2,X) = (A,X,1)
∂(2,a) = (3,X,1)
∂(2,b) = (7,X,1)
∂(3,a) = (3,a,1)
∂(3,b) = (3,b,1)

∂(3,_) = (4,_,0)
∂(3,X) = (4,X,0)
∂(4,a) = (5,X,0)
∂(5,a) = (5,a,0)
∂(5,b) = (5,b,0)
∂(5,X) = (2,X,1)
∂(6,b) = (5,X,0)

∂(7,a) = (7,a,1)
∂(7,b) = (7,b,1)
∂(7,_) = (6,_,0)
∂(7,X) = (6,X,0)

For any other:
∂(q,s) = (R,s,1)

(1, >abbbba, 0)
⇒ (2, >abbbba, 1)
⇒ (3, >Xbbbba, 2)
⇒ (3, >Xbbbba, 3)
⇒ (3, >Xbbbba, 4)
⇒ (3, >Xbbbba, 5)
⇒ (3, >Xbbbba, 6)
⇒ (3, >Xbbbba_, 7)
⇒ (4, >Xbbbba_, 6)
⇒ (5, >XbbbbX_, 5)
⇒ (5, >XbbbbX_, 4)
⇒ (5, >XbbbbX_, 3)
⇒ (5, >XbbbbX_, 2)
⇒ (5, >XbbbbX_, 1)
⇒ (2, >XbbbbX_, 2)
⇒ (7, >XXbbbX_, 3)
⇒ (7, >XXbbbX_, 4)
⇒ …
…
⇒ (A, >XXXXXX_, 5)
→ accepting configuration

Language of a Turing machine

Turing machine M **accepts** string u if $(q_s, >u, 0) \Rightarrow^* (q_{acc}, v, i)$ for some v, i

Turing machine M **rejects** string u if $(q_s, >u, 0) \Rightarrow^* (q_{rej}, v, i)$ for some v, i

The **language accepted** by Turing machine M is:

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

Turing-computable languages

A Turing machine M is total if for every u $\in \Sigma^*$, M accepts u or M rejects u

- A total Turing machine has to halt on every input

Language A is Turing-computable if there exists a total Turing machine M such that L(M) = A

- i.e., if there exists a Turing machine that accepts every string in A and rejects every string not in A

Turing-computability

Every regular (= FA-computable) language is Turing-computable

- Regular = can be accepted by some finite state machine, so convert the finite state machine to a Turing machine (easy)

All these languages are Turing-computable:

$\{a^n b^n \mid n \geq 0\}$

$\{a^n b^m \mid m \geq n \geq 0\}$

$\{a^n b^n c^n \mid n \geq 0\}$

$\{u \in \{a,b\}^* \mid u = rev(u)\}$

$\{u \in \{a,b\}^* \mid \#_a(u) = \#_b(u) \}$

Natural number arithmetic is Turing-computable

$\{ n_1\#n_2\#n_3 \mid n_1 + n_2 = n_3 \}$ is Turing-computable

$\{ n_1\#n_2\#n_3 \mid n_1\ n_2 = n_3 \}$ is Turing-computable

$\{ n_1\#n_2 \mid n_1 \text{ divides } n_2 \}$ is Turing-computable

$\{ n \mid n \text{ is prime} \}$ is Turing-computable

Claim: Turing-computability is a reasonable definition of computability