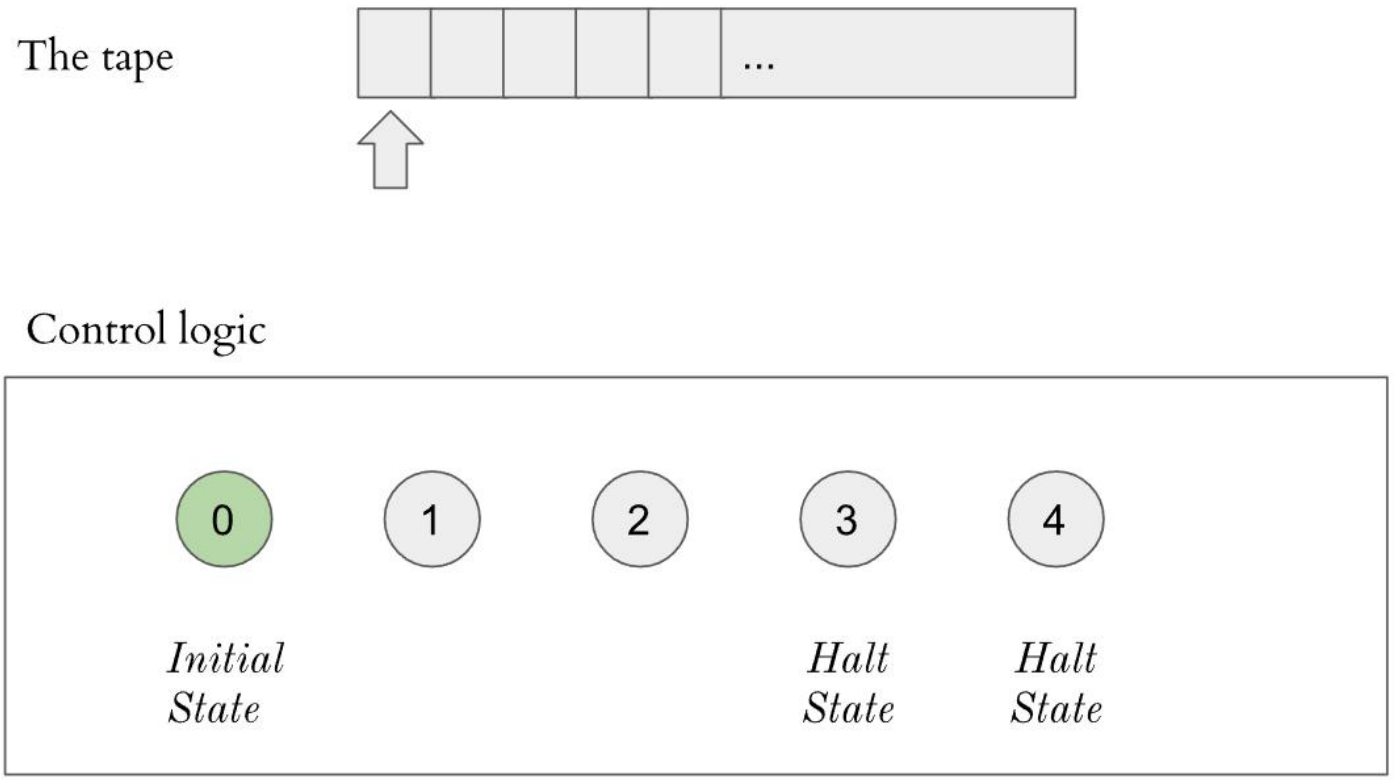
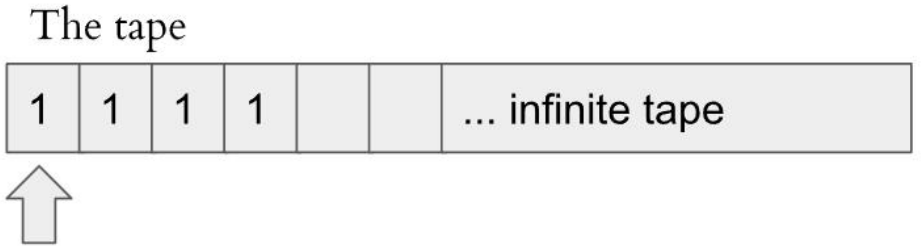
The Execution of A Turing Machine

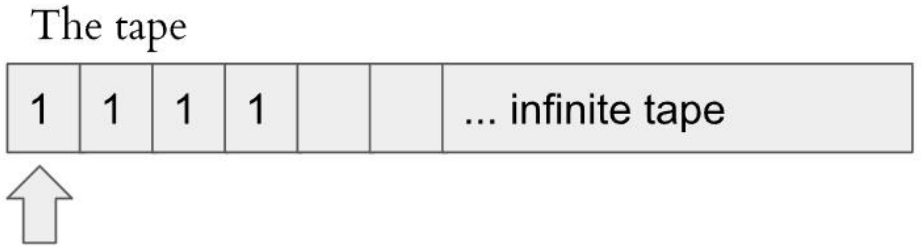

The tape
...
Control logic
0
1
2
3
4
Initial State
Halt State
Halt State

The tape is to be initialized by a finite binary string.

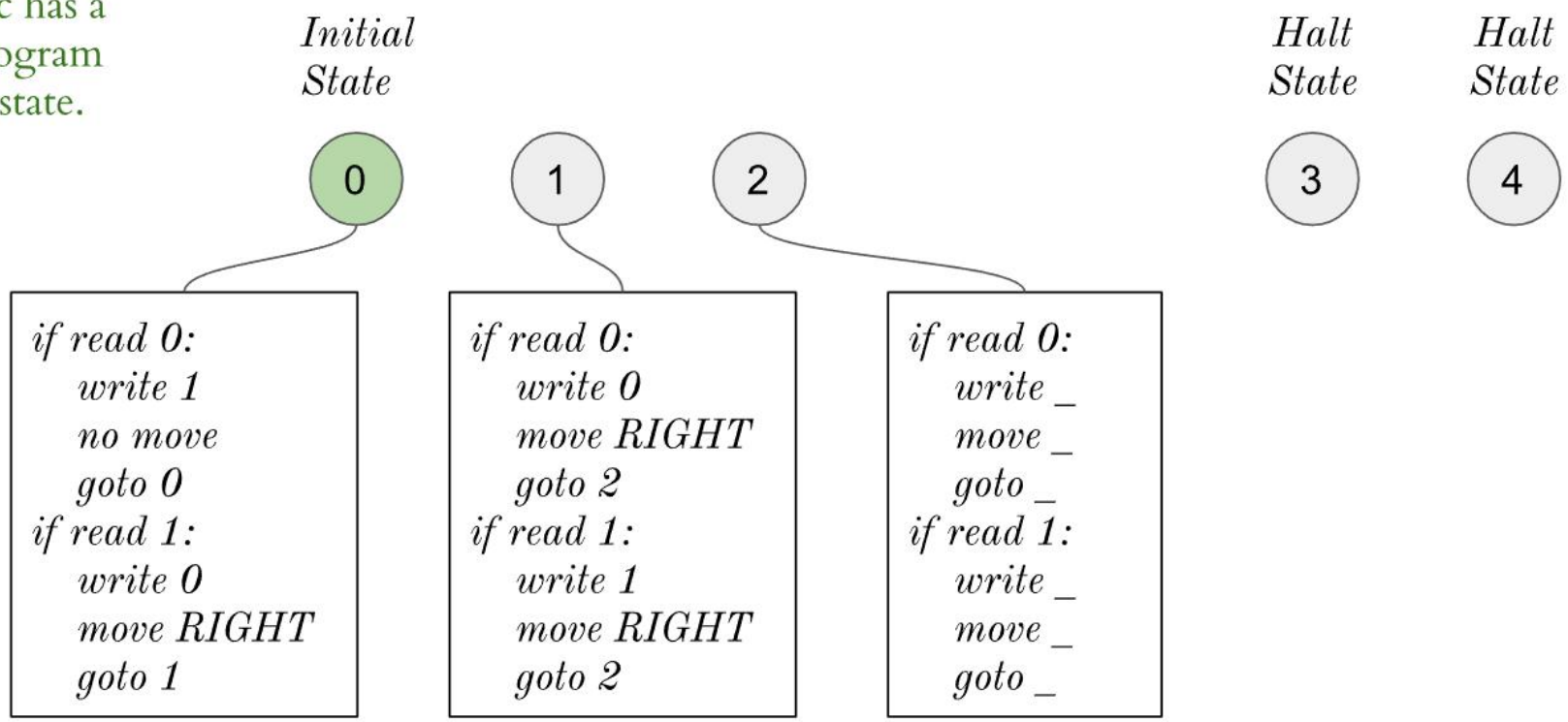


The control logic has a simple switch program for *each non-halt* state.

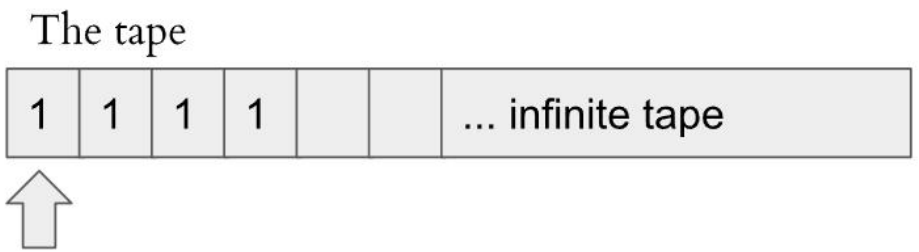
The tape is to be initialized by a finite binary string.



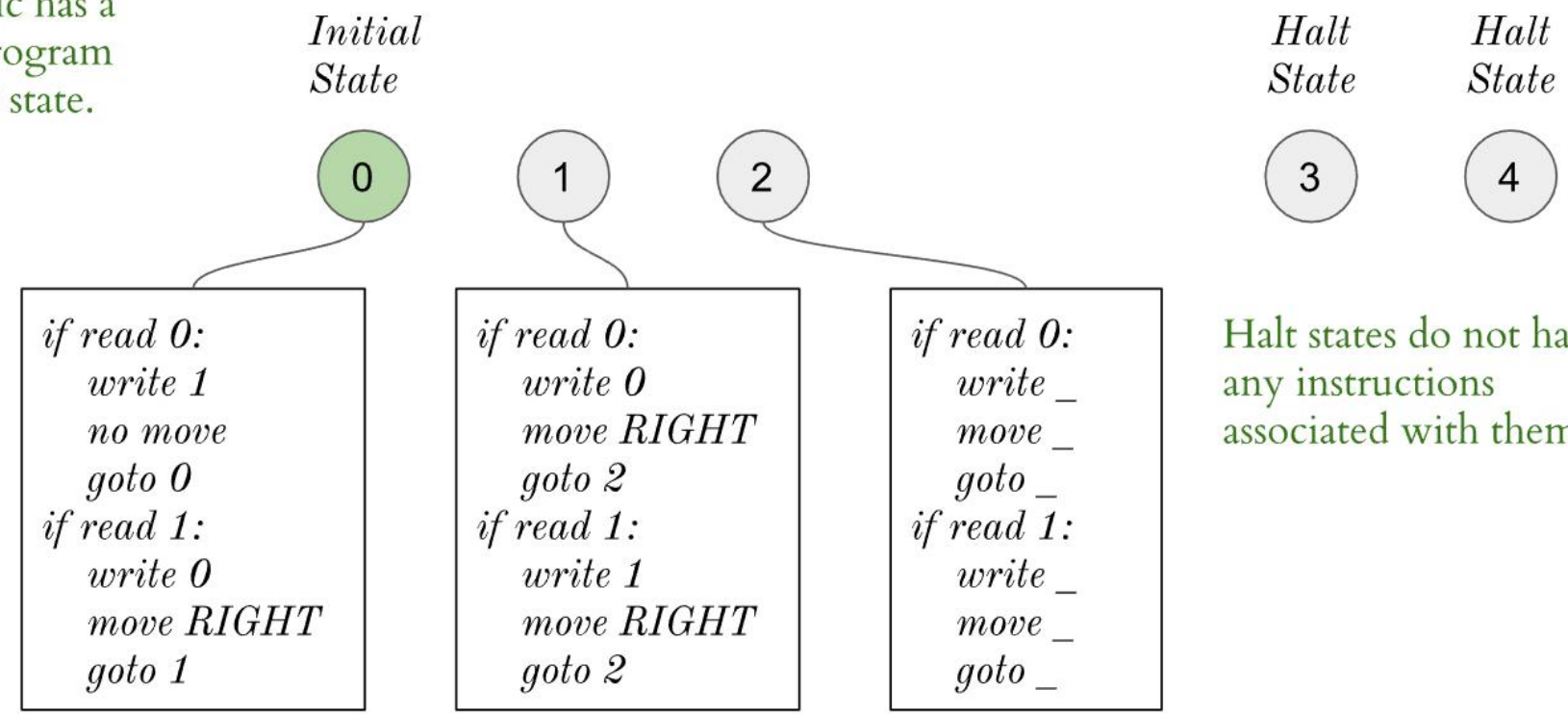
The control logic has a simple switch program for *each non-halt* state.



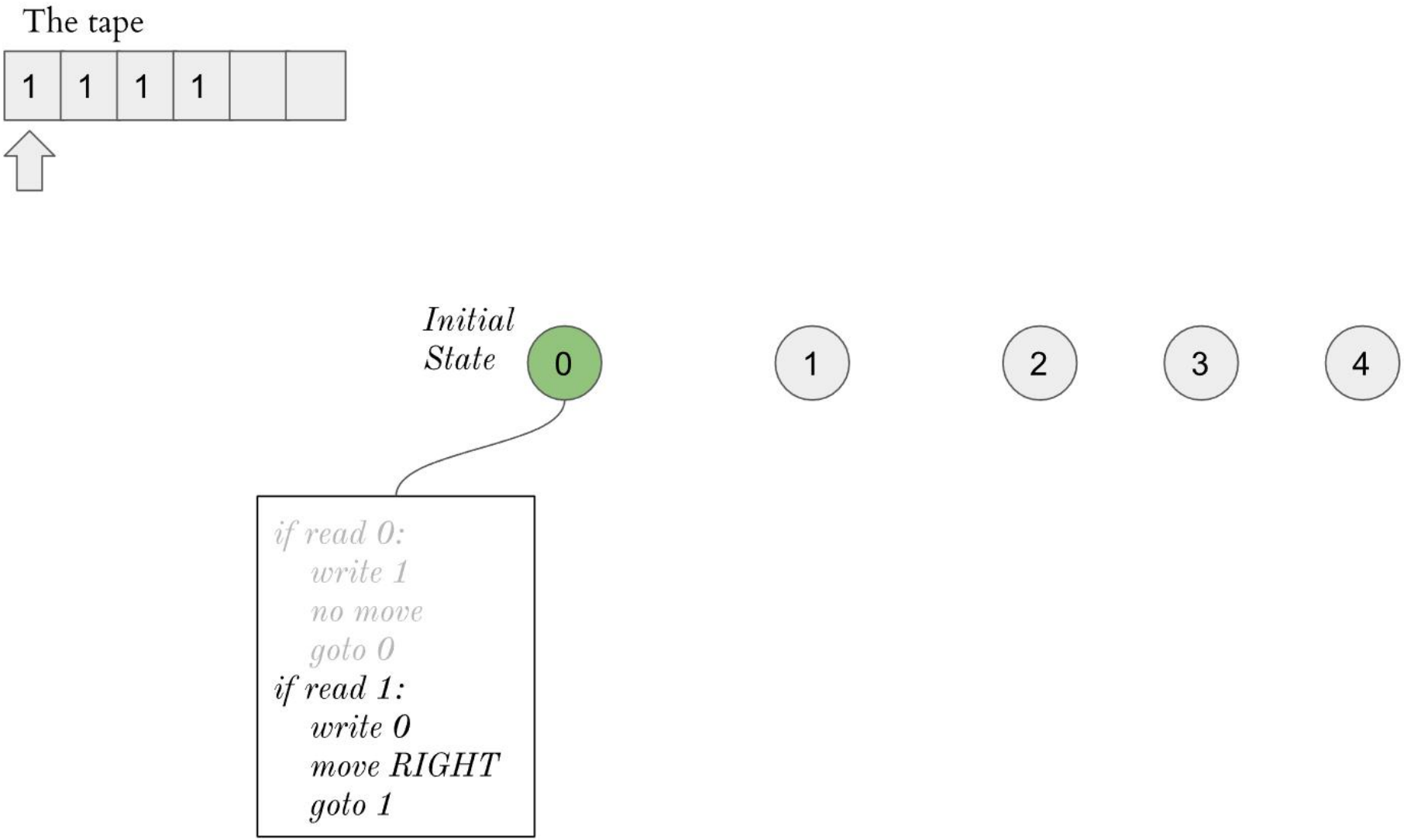
The tape is to be initialized by a finite binary string.

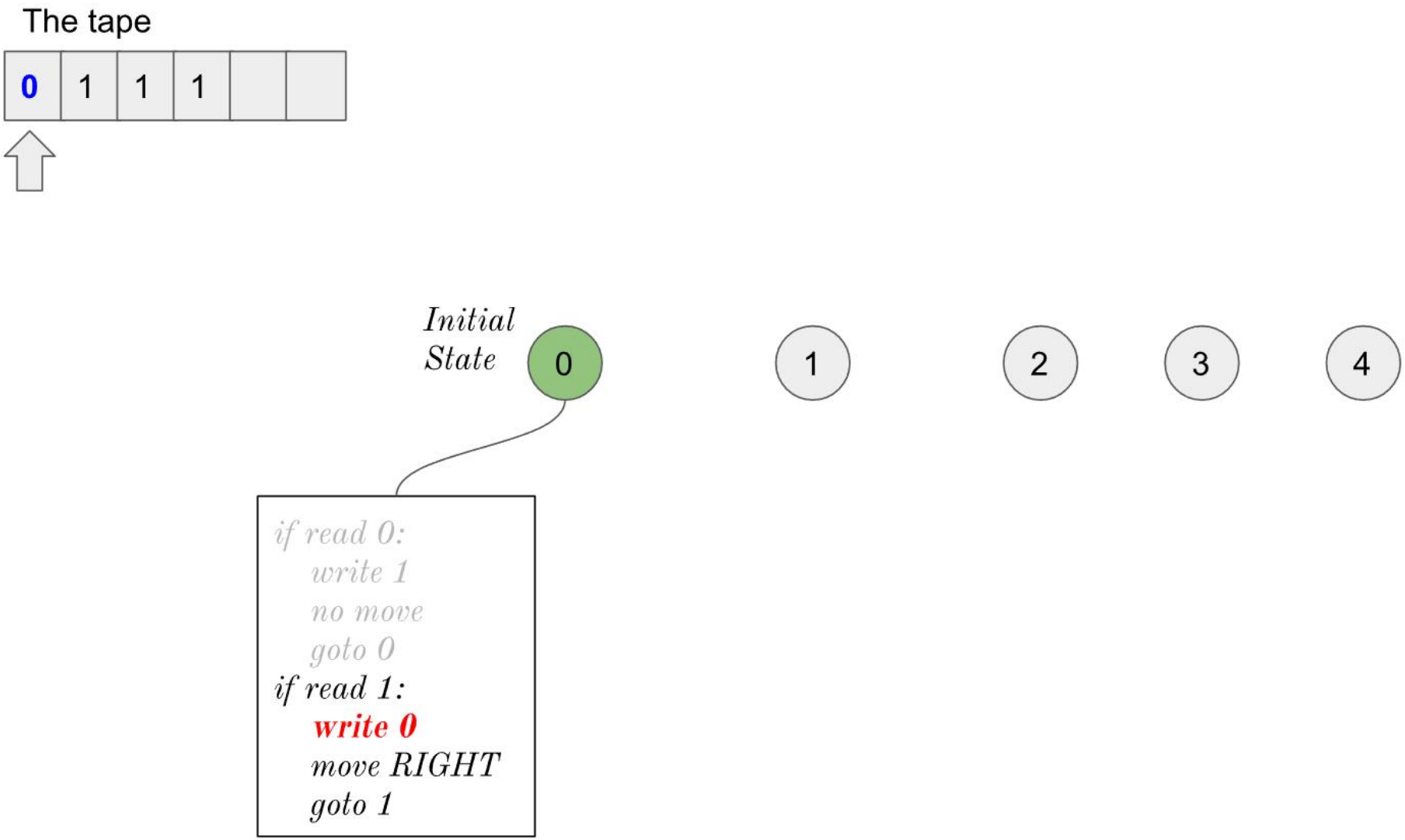


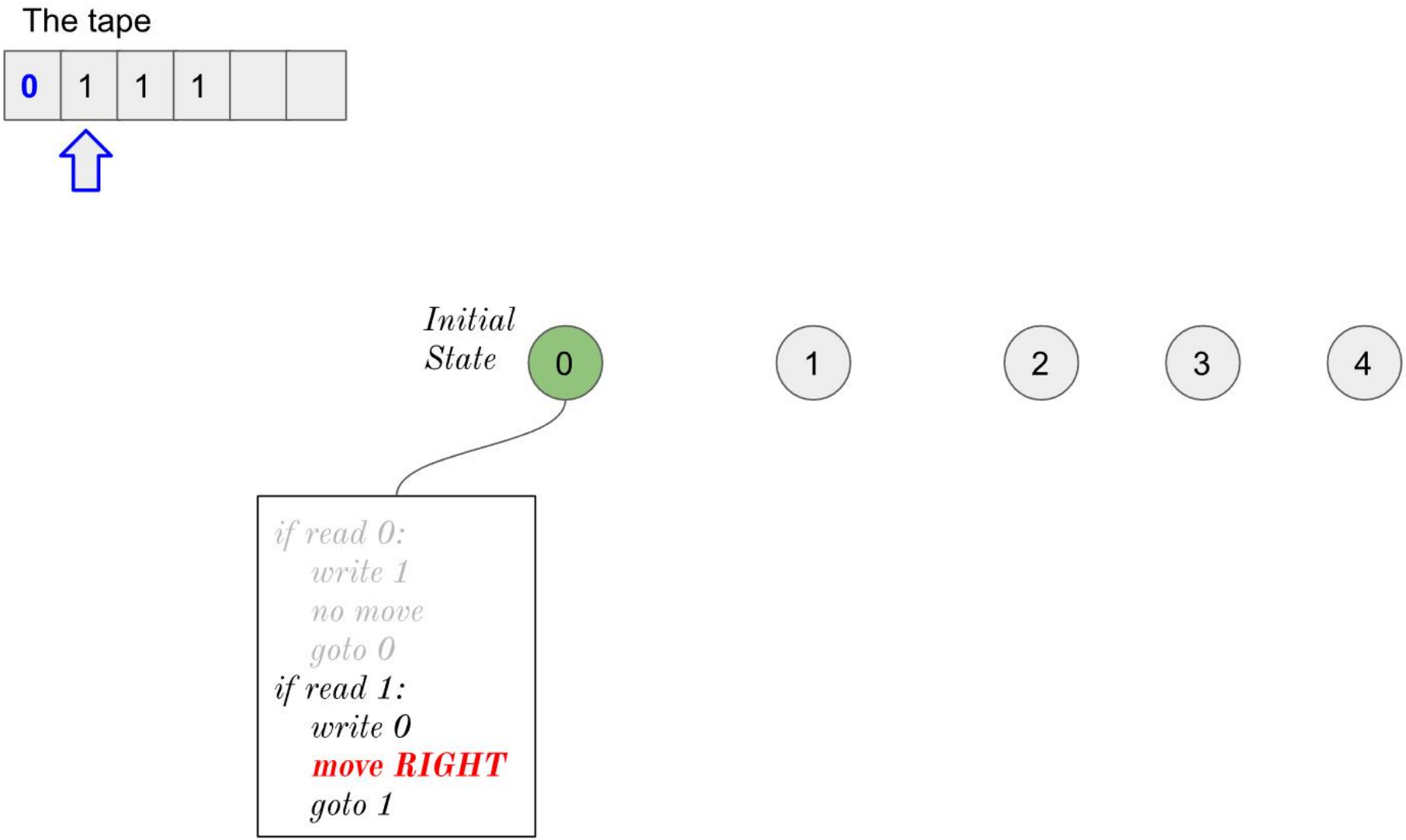
The control logic has a simple switch program for *each non-halt* state.



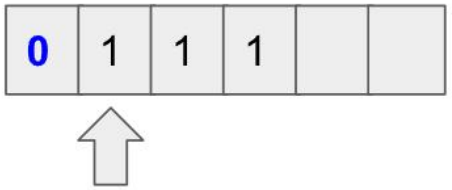
Halt states do not have any instructions associated with them


The tape
1
1
1
1
Initial State
0
1
2
3
4
if read 0:
write 1
no move
goto 0
if read 1:
write 0
move RIGHT
goto 1


The tape
0
1
1
1
Initial State
0
1
2
3
4
if read 0:
write 1
no move
goto 0
if read 1:
write 0
move RIGHT
goto 1

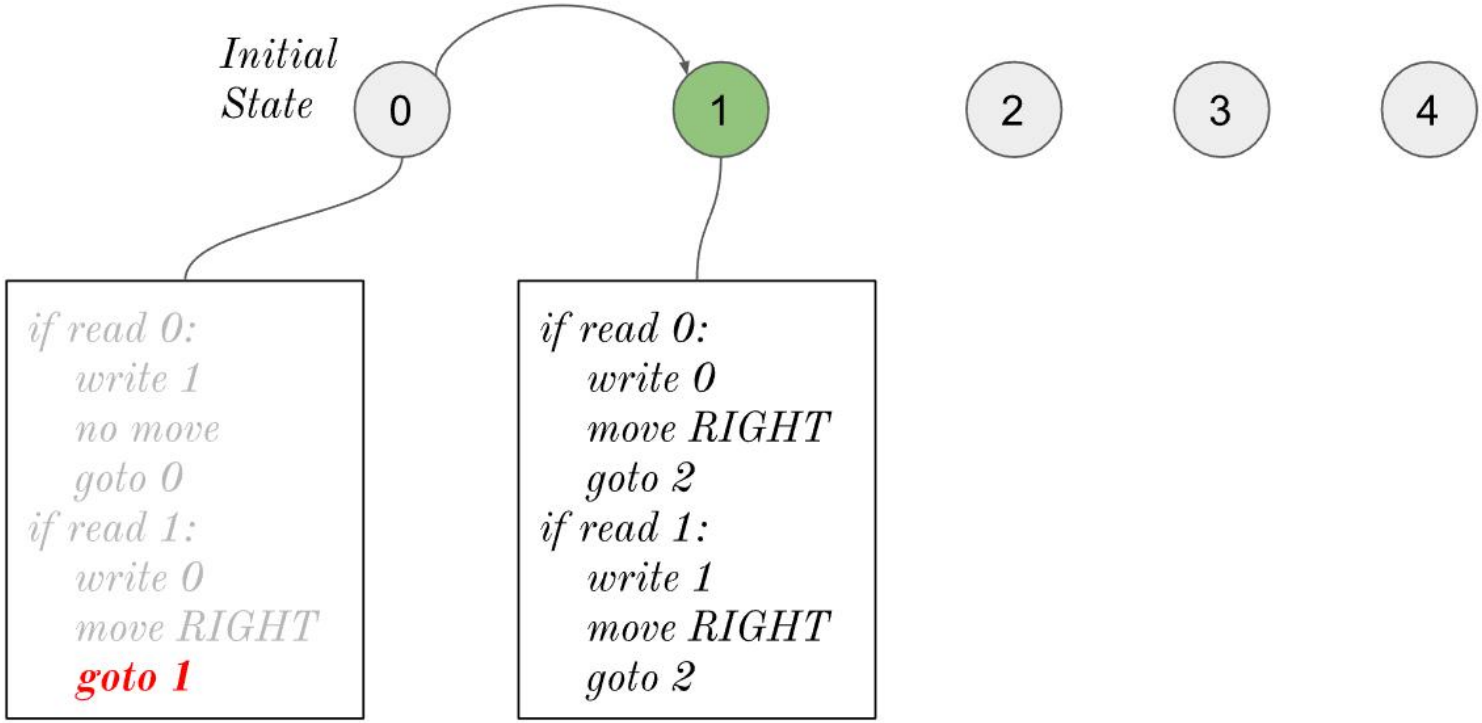

The tape
0
1
1
1
Initial State
0
1
2
3
4
if read 0:
write 1
no move
goto 0
if read 1:
write 0
move RIGHT
goto 1

The tape



Challenge:

Work out the execution of the TM for state 1 and the current tape configuration.



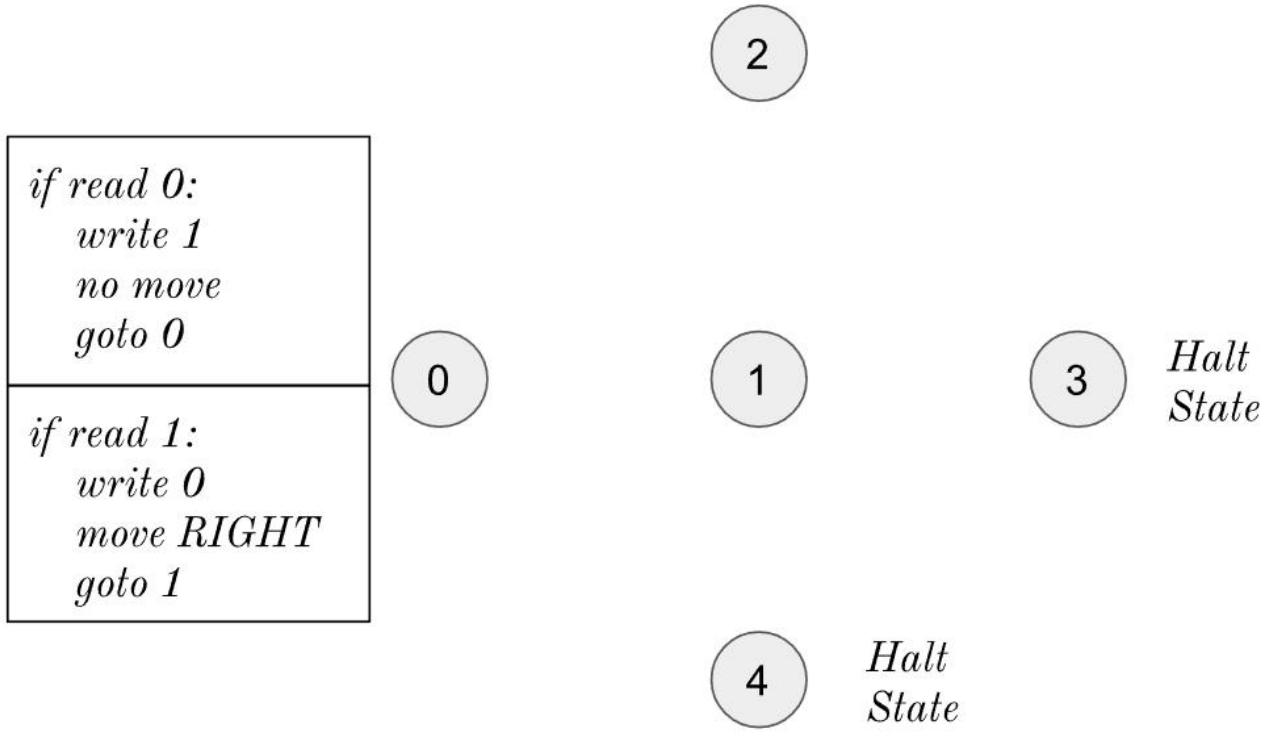
The TM executes by:

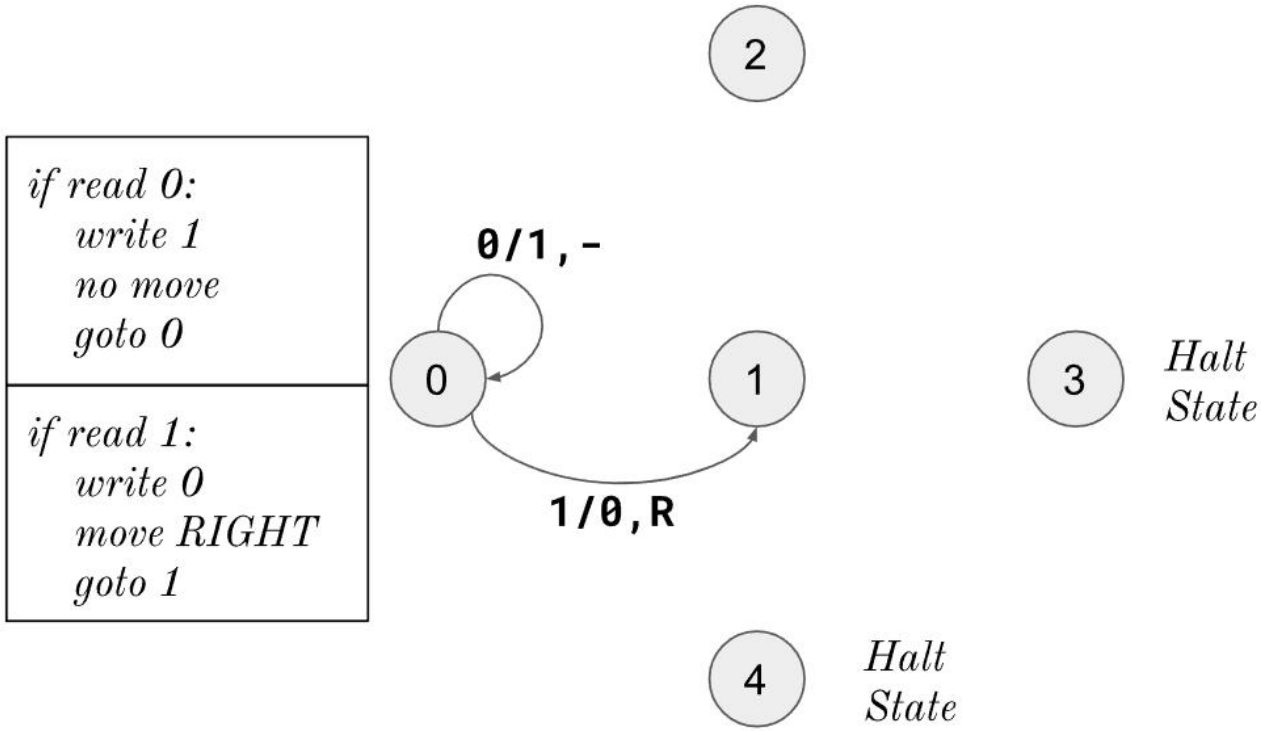
1. Read symbol from the *tape* at the *current* position.
2. Execute the instructions of the *current state*.
3. If state is *halt*, stop
   Else repeat.

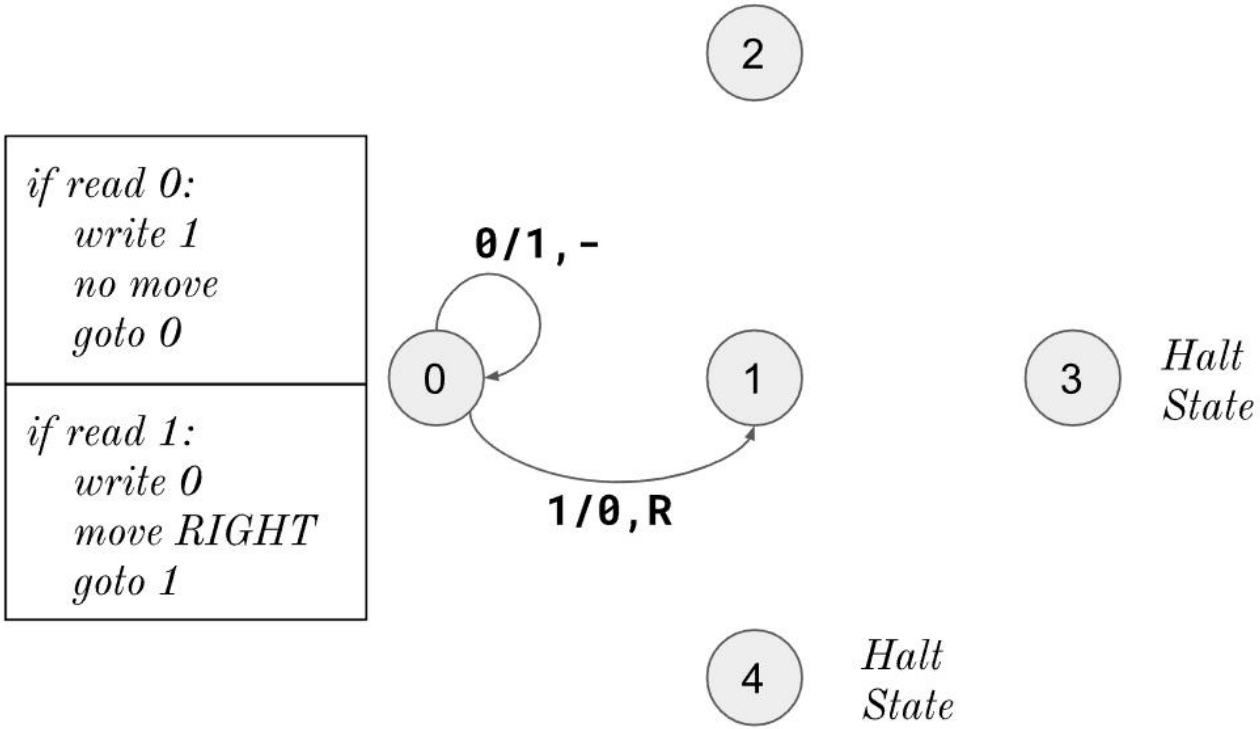
The execution of a TM:

1. It is deterministic.
2. Termination is *not guaranteed*.

Transitional Diagrams for TM

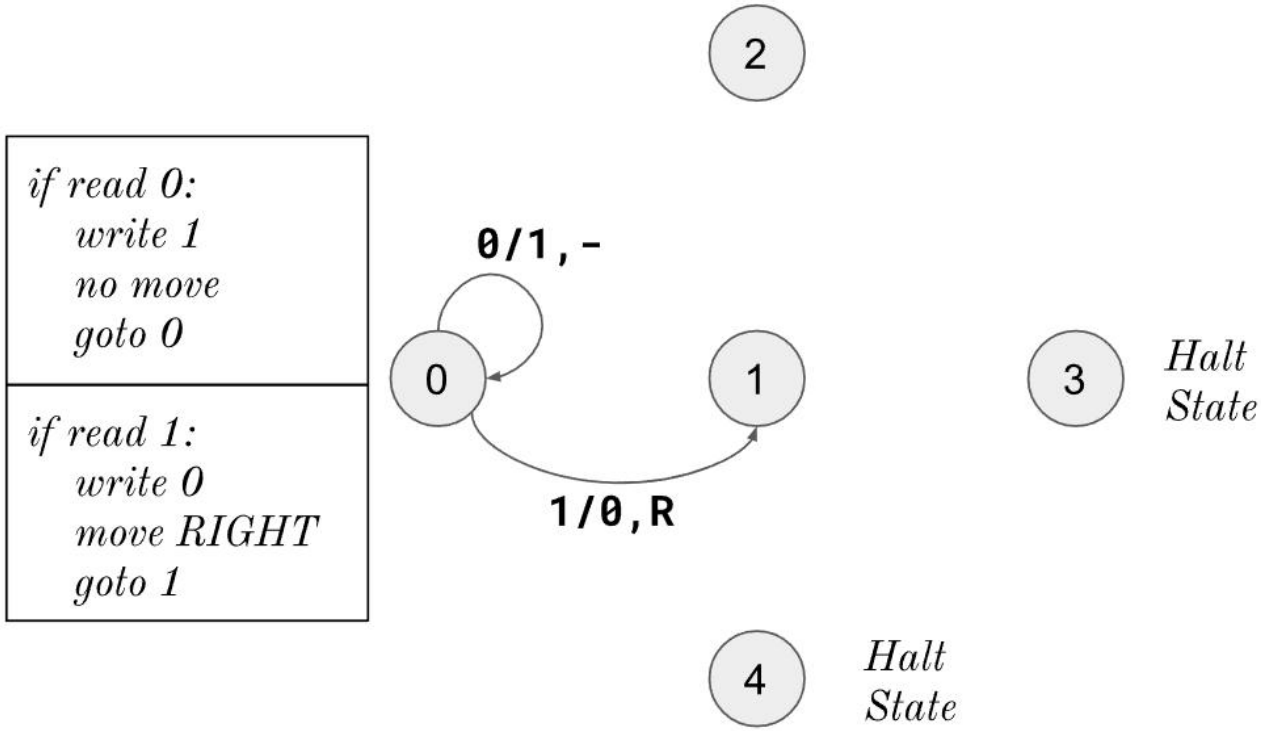

2
if read 0:
write 1
no move
goto 0
0
1
3
Halt
State
if read 1:
write 0
move RIGHT
goto 1
4
Halt
State


2
if read 0:
write 1
no move
goto 0
0/1,-
0
1
3
Halt
State
if read 1:
write 0
move RIGHT
goto 1
1/0,R
4
Halt
State

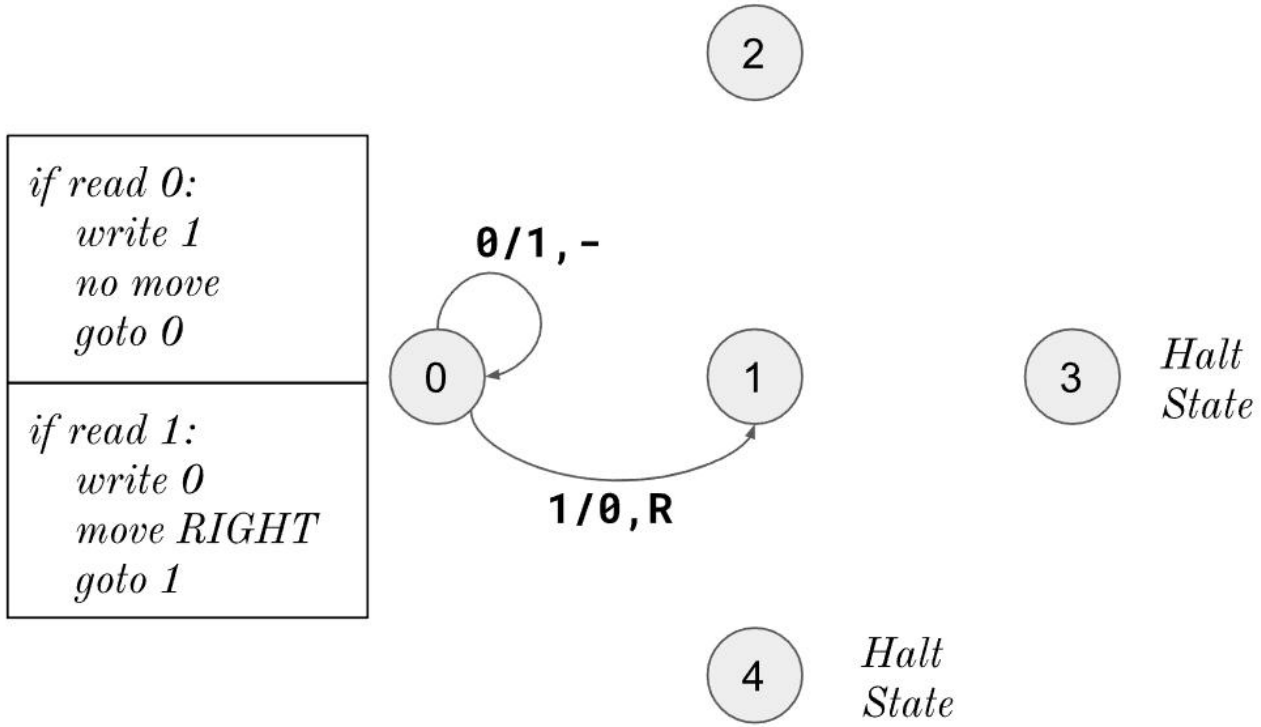


if read 0
write 1
no move

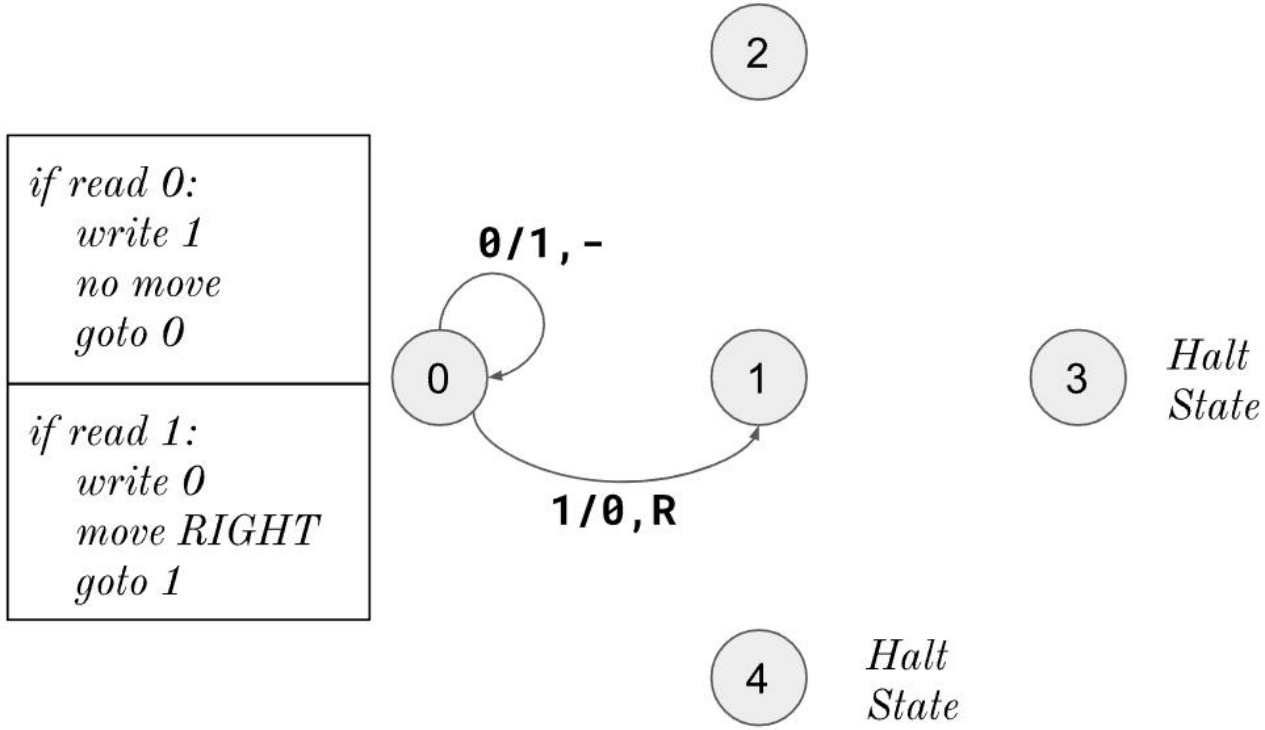
[ ] / [ ] , [ ]



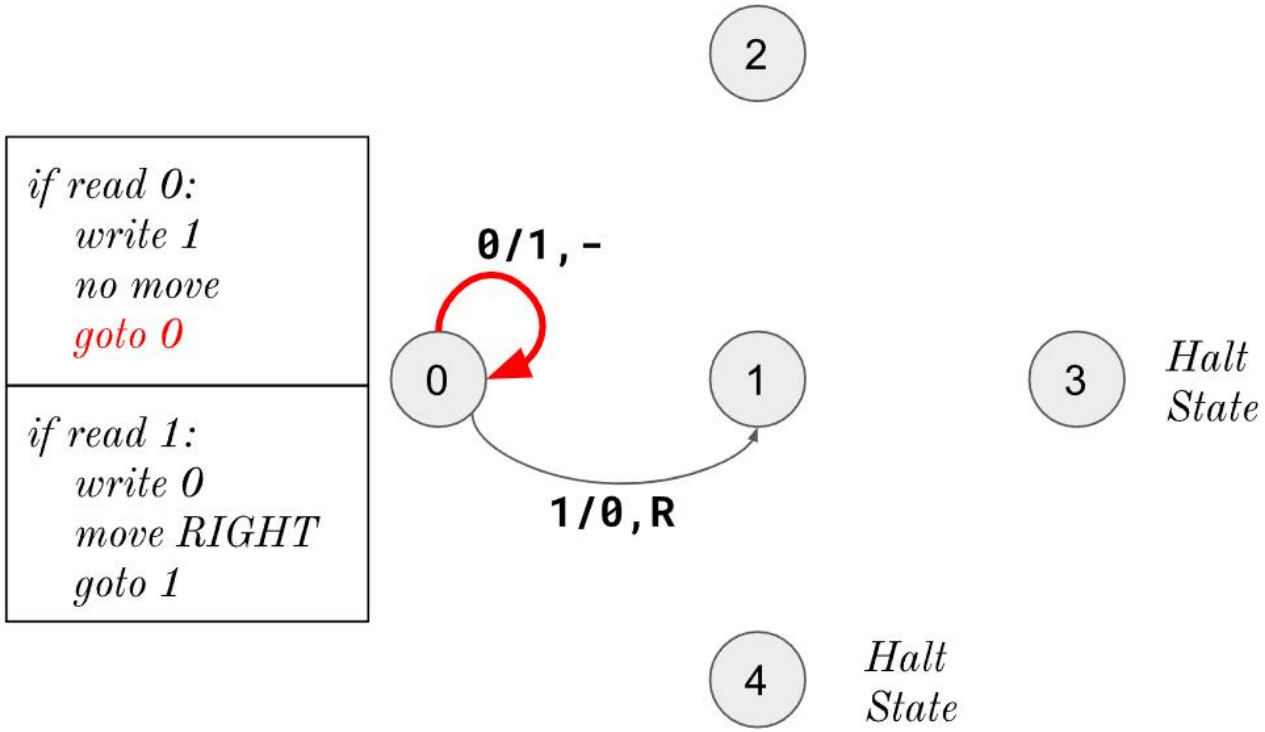
if read 0
write 1 **0 / 1 , -**
no move



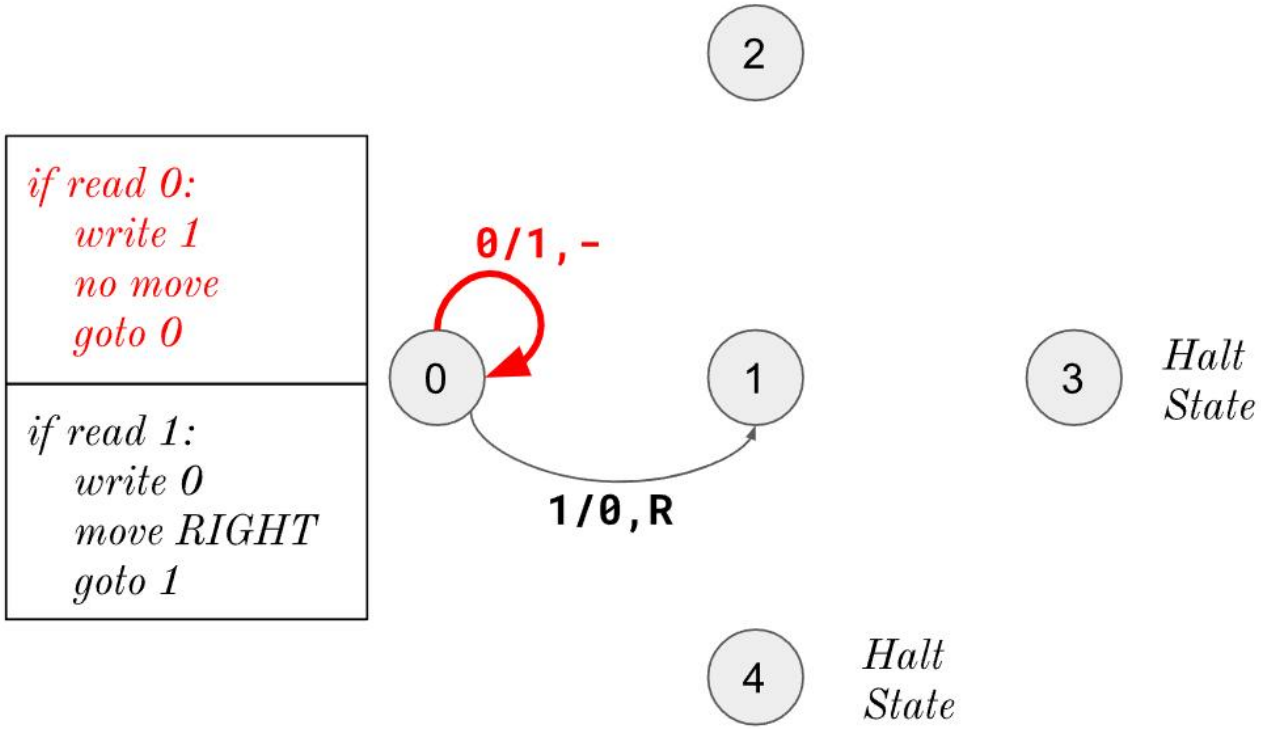
if read 0
write 1 **0 / 1 , -**
no move

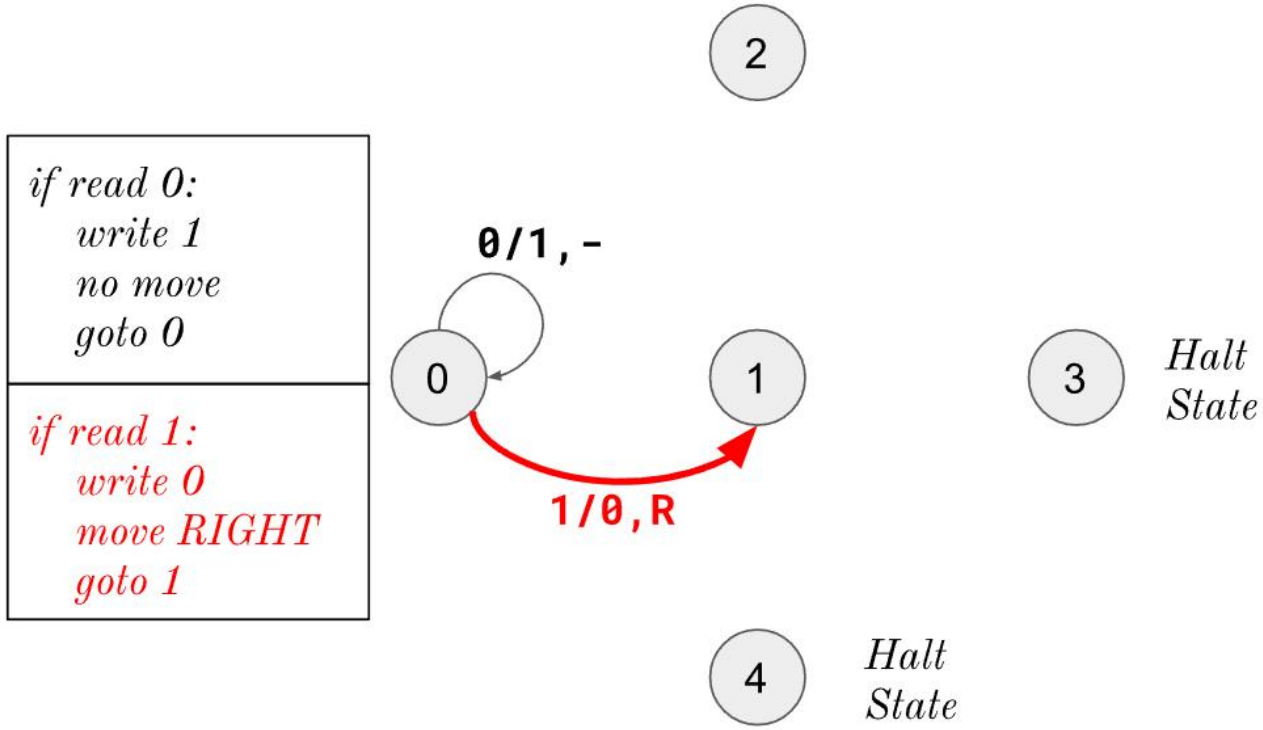


if read 0
write 1 **0 / 1 , -**
no move



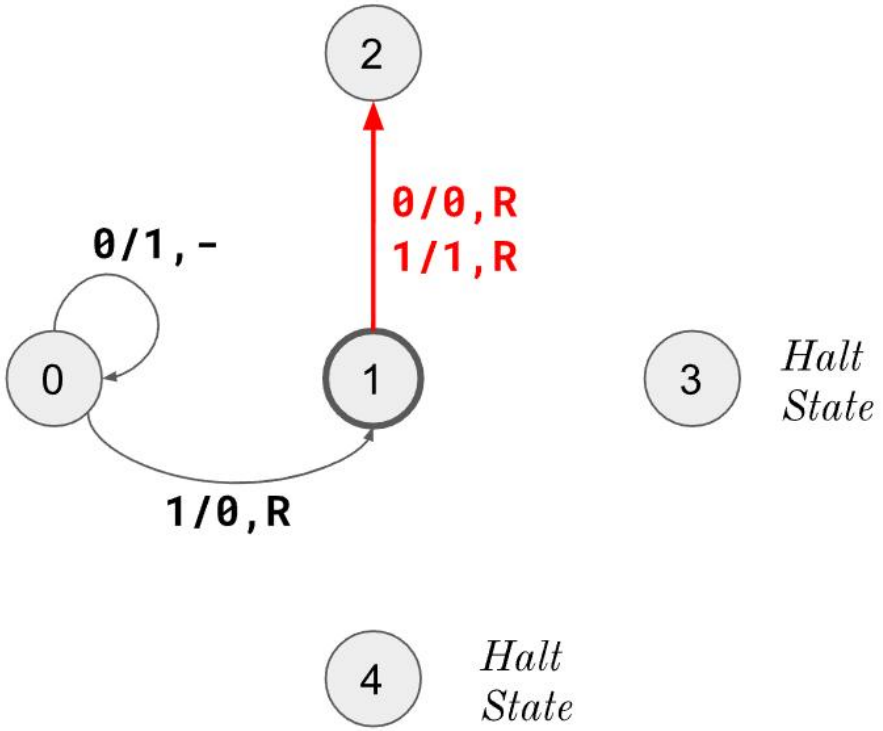
if read 0
write 1 **0 / 1 , -**
no move


2
if read 0:
write 1
no move
goto 0
0/1,-
0
1
3
Halt
State
if read 1:
write 0
move RIGHT
goto 1
1/0,R
4
Halt
State


2
if read 0:
write 1
no move
goto 0
0/1,-
0
1
3
Halt
State
if read 1:
write 0
move RIGHT
goto 1
1/0,R
4
Halt
State

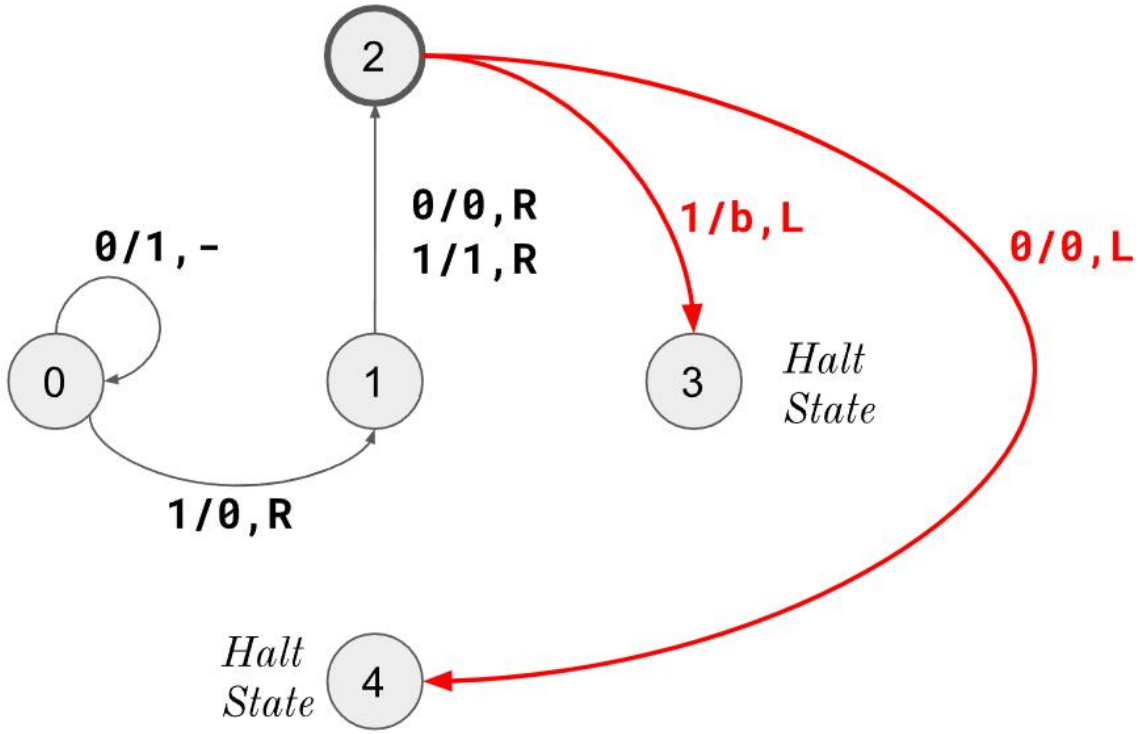
Let's encode the instructions for state 1.

```
if read 0:
    write 0
    move RIGHT
    goto 2
if read 1:
    write 1
    move RIGHT
    goto 2
```



What is the instructions for state 2?

if read 0:
write __
move __
goto __
if read 1:
write __
move __
goto __



1	1	1	1	0	0	...

Challenge:

Complete the execution of this TM with the given input tape?