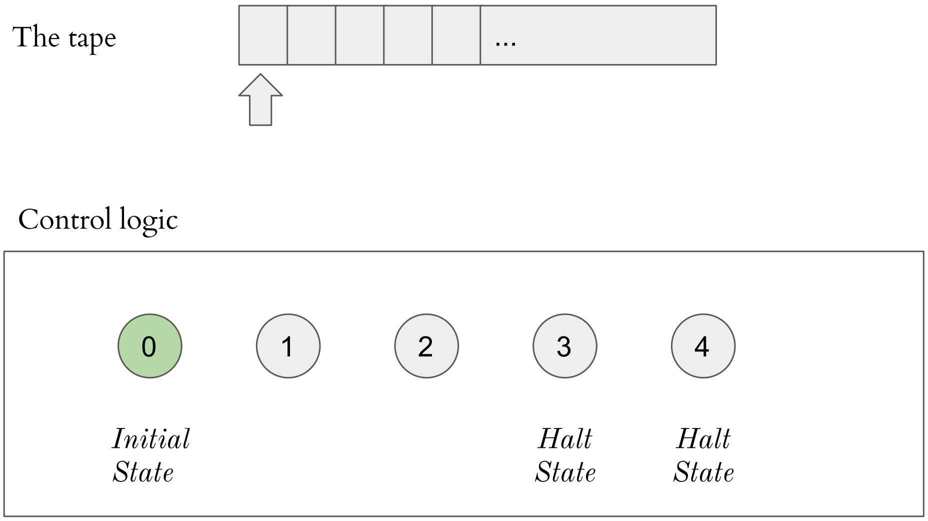
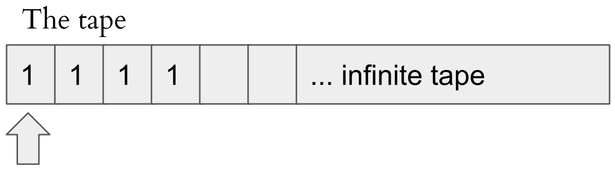
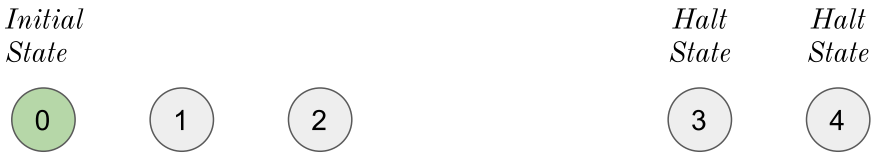
The Execution of A Turing Machine



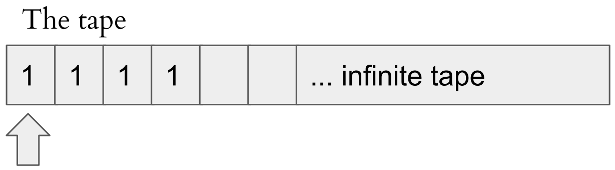
The tape is to be initialized by a finite binary string.



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

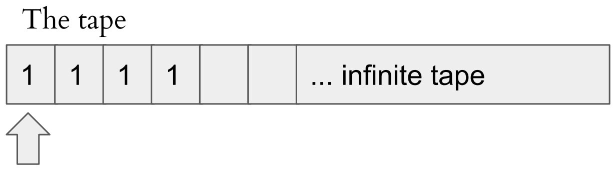


The tape is to be initialized by a finite binary string.

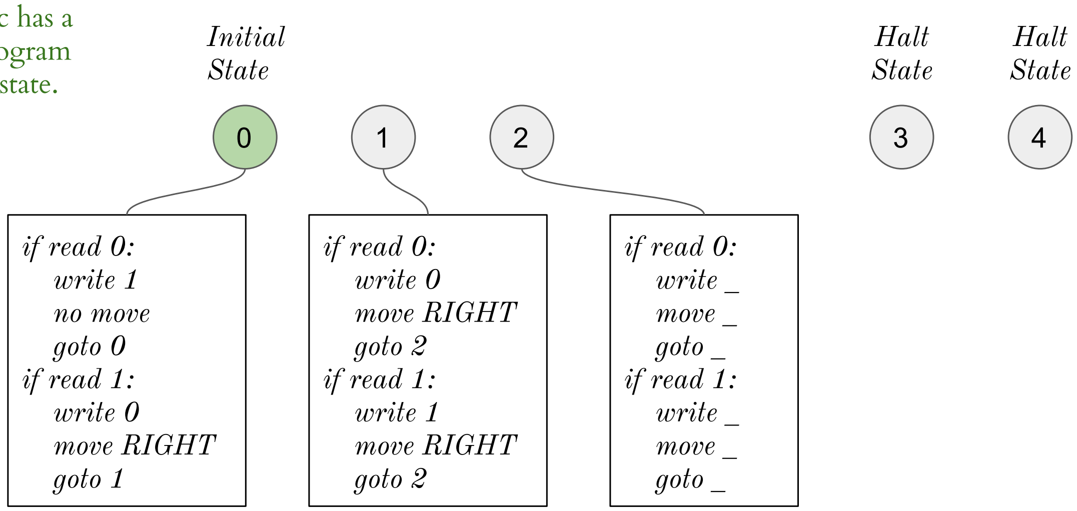


The control logic has a simple switch program for *each non-halt* state.

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



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

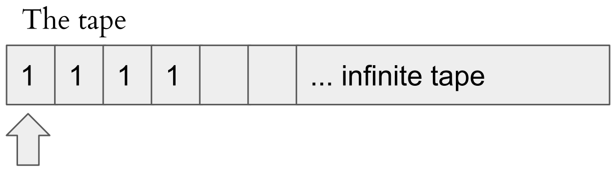


```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```

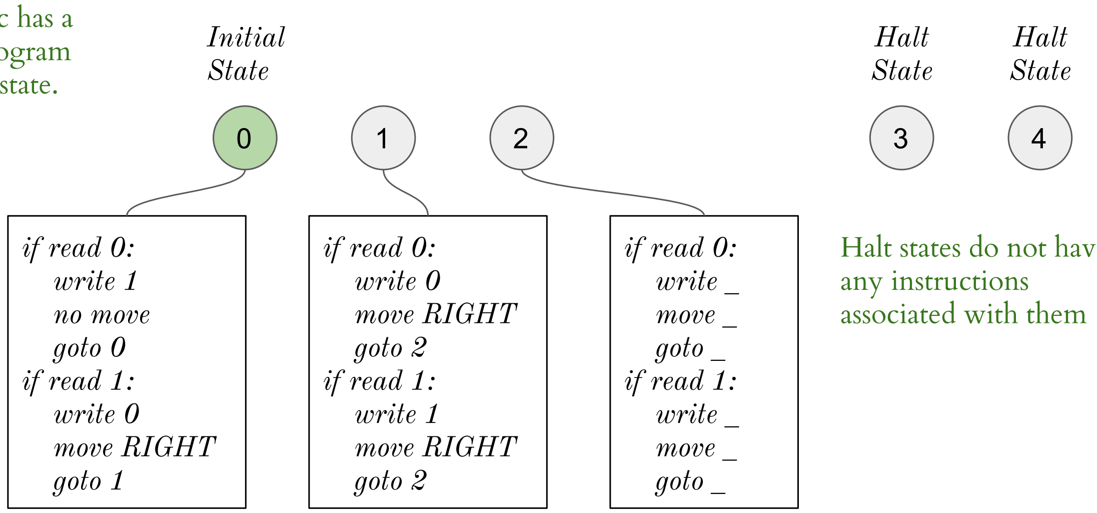
```
if read 0:
    write 0
    move RIGHT
    goto 2
if read 1:
    write 1
    move RIGHT
    goto 2
```

```
if read 0:
    write _
    move _
    goto _
if read 1:
    write _
    move _
    goto _
```

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



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

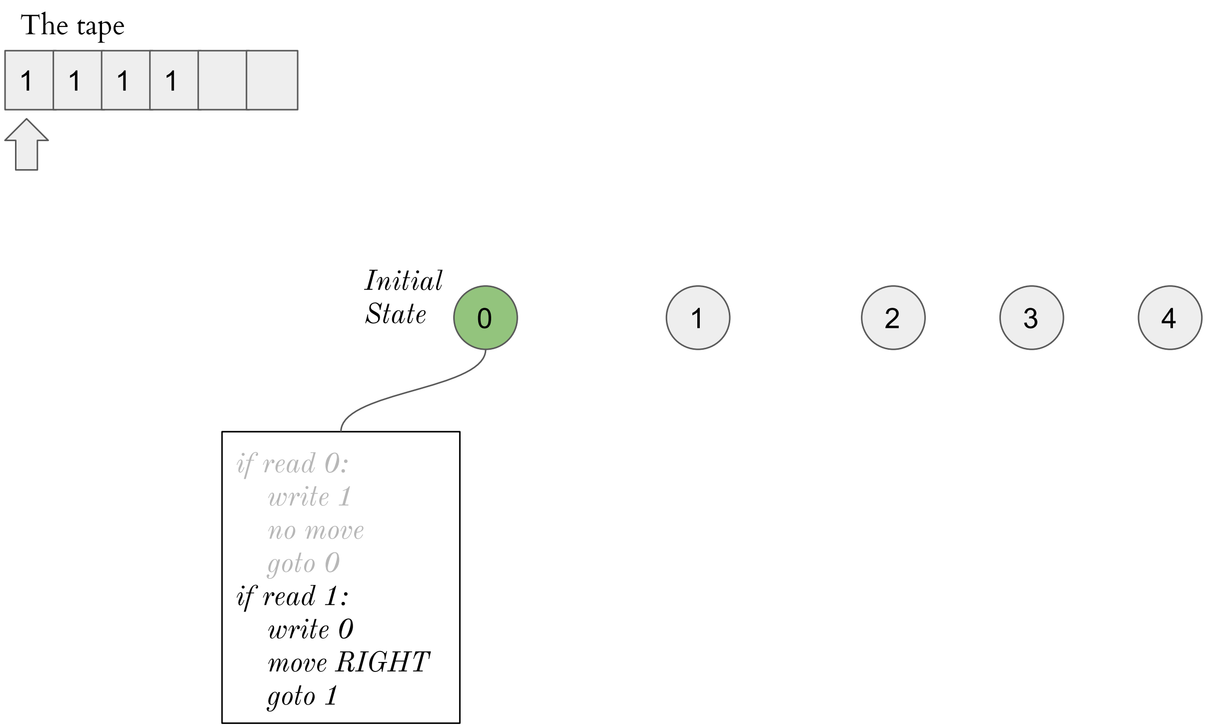


```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```

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

```
if read 0:
    write _
    move _
    goto _
if read 1:
    write _
    move _
    goto _
```

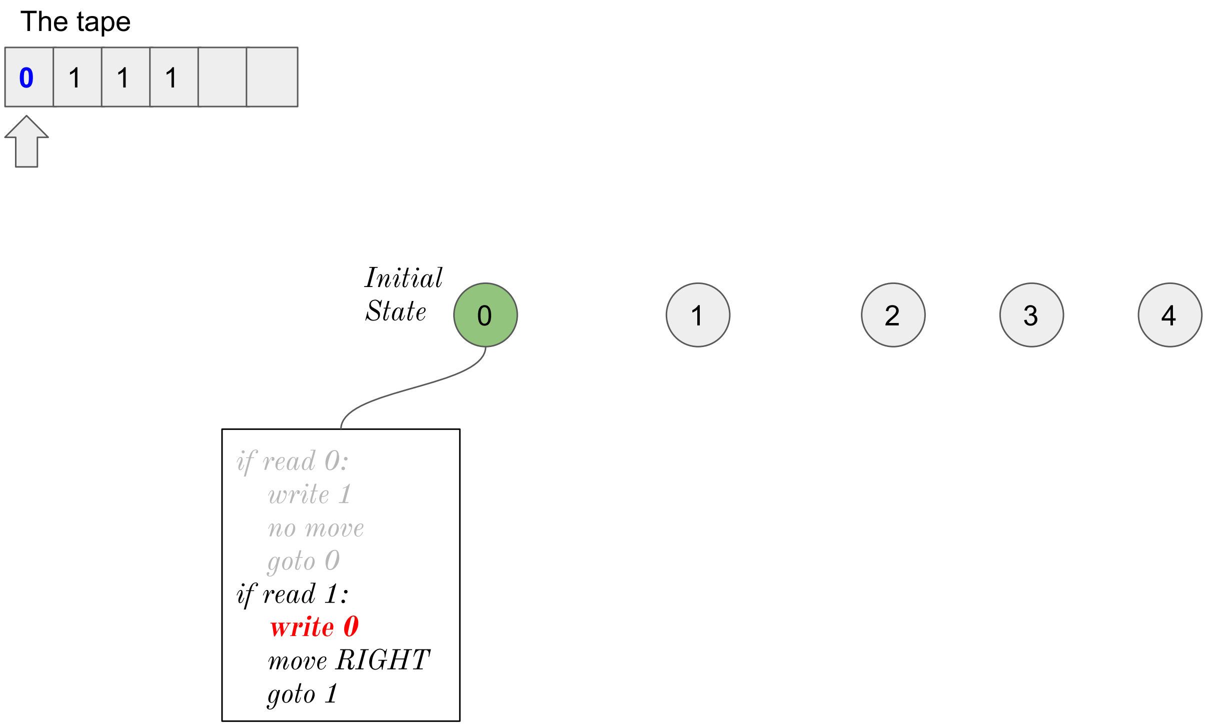
Halt states do not have any instructions associated with them



The tape

1	1	1	1		

```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```

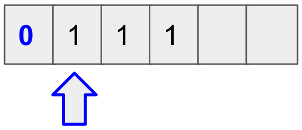


The tape

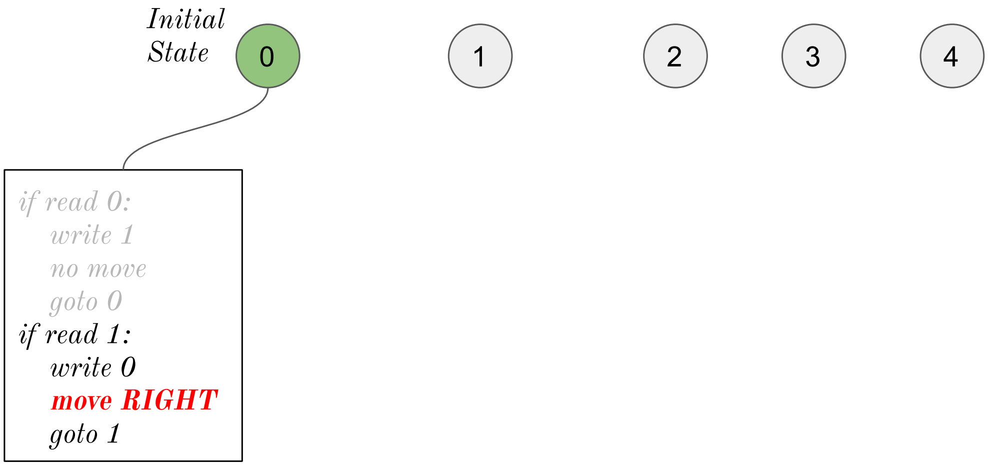
0	1	1	1		

```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```

The tape

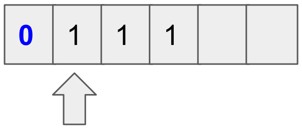


0	1	1	1		



```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```

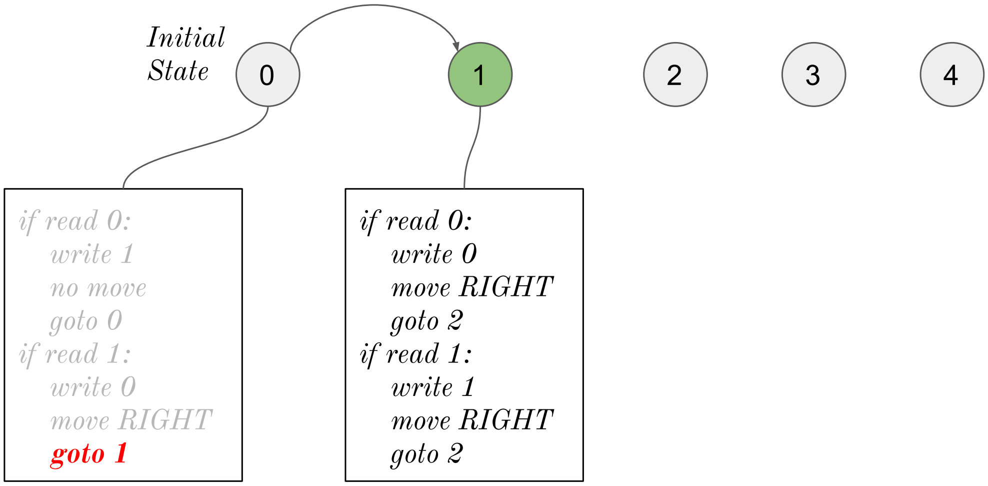
The tape



0	1	1	1		

Challenge:

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



```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```

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

The TM executes by:

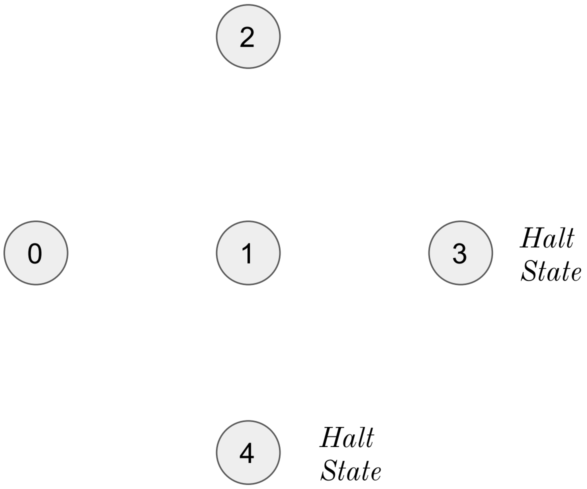
1. Read symbol from the *tape* at the *current* position.
2. Execute the instructions of the *current state*.
3. If state is *halt*, stop
   Else repeat.

The execution of a TM:

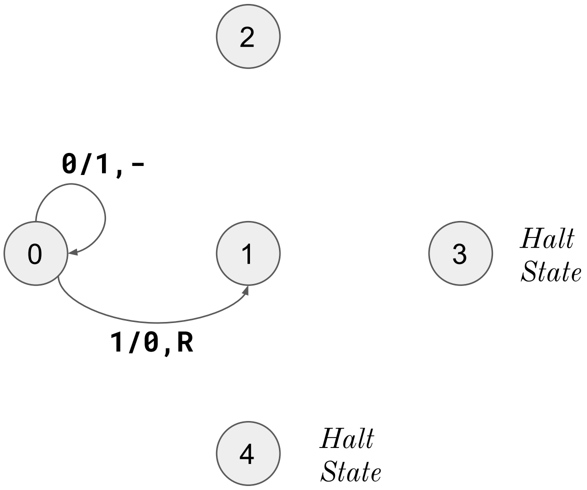
1. It is deterministic.
2. Termination is *not guaranteed*.

Transitional Diagrams for TM

```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```

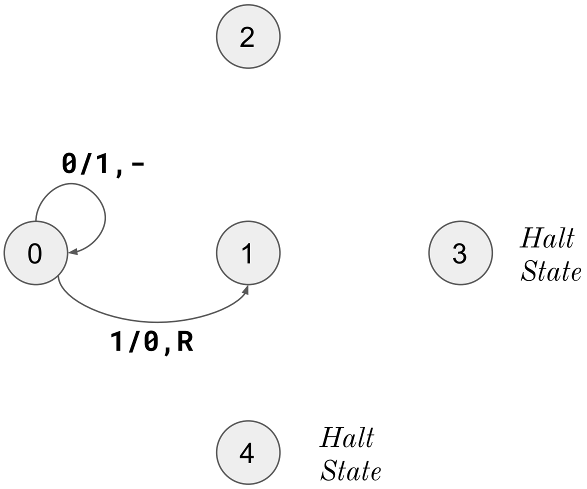


```
if read 0:
    write 1
    no move
    goto 0
if read 1:
    write 0
    move RIGHT
    goto 1
```



```
if read 0:
    write 1
    no move
    goto 0
```

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

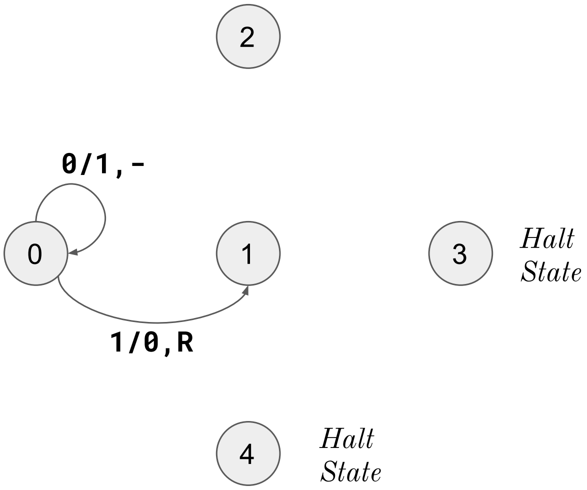


```
if read 0
    write 1
    no move
```

□ / □ , □

```
if read 0:
    write 1
    no move
    goto 0
```

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

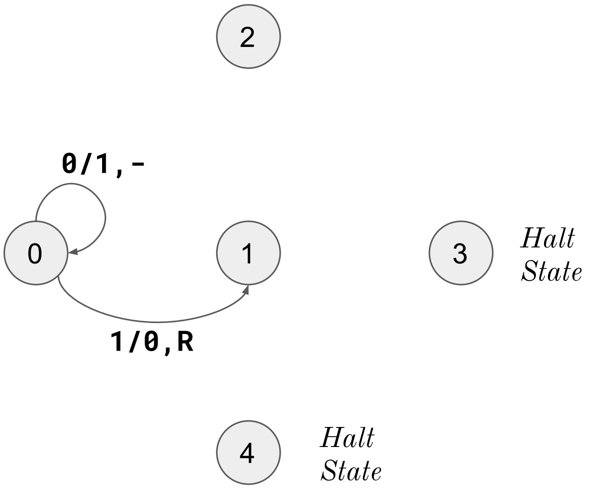


```
if read 0
    write 1
    no move
```

0 / 1 , -

```
if read 0:
    write 1
    no move
    goto 0
```

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

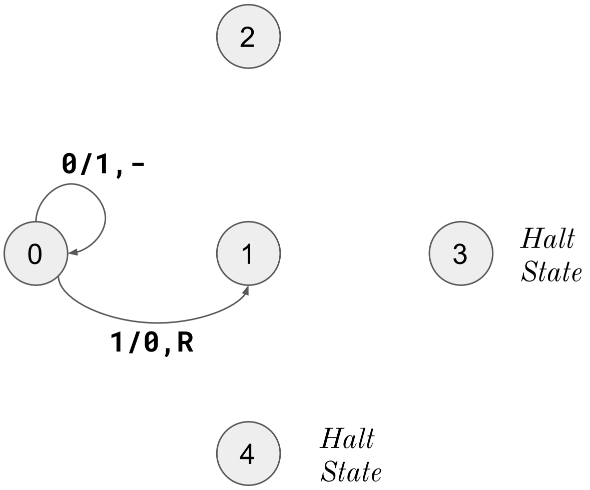


if read 0
  write 1
  no move

0 / 1 , -

```
if read 0:
    write 1
    no move
    goto 0
```

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

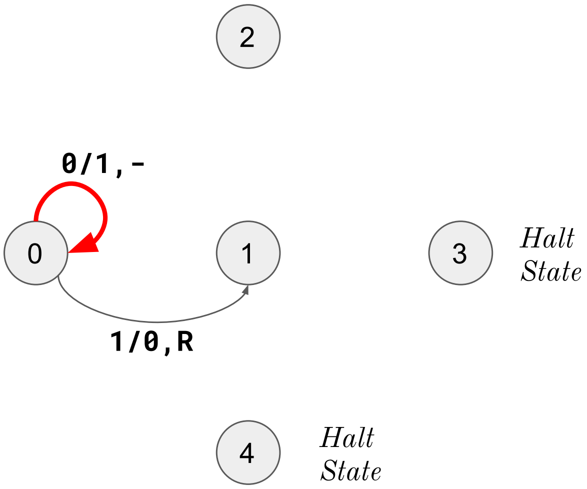


if read 0
  write 1
  no move

0 / 1 , -

```
if read 0:
    write 1
    no move
    goto 0
```

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

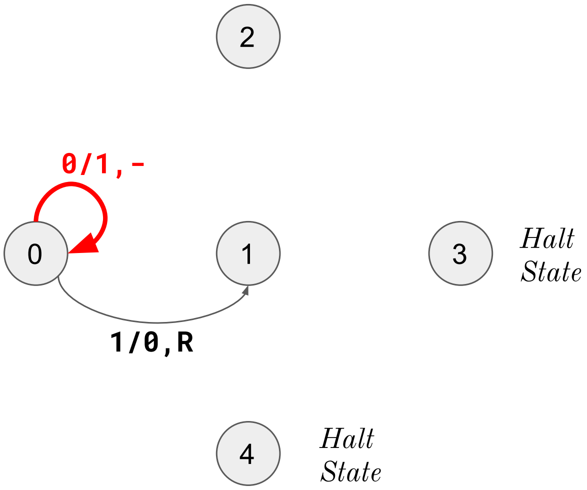


```
if read 0
    write 1
    no move
```

0 / 1 , -

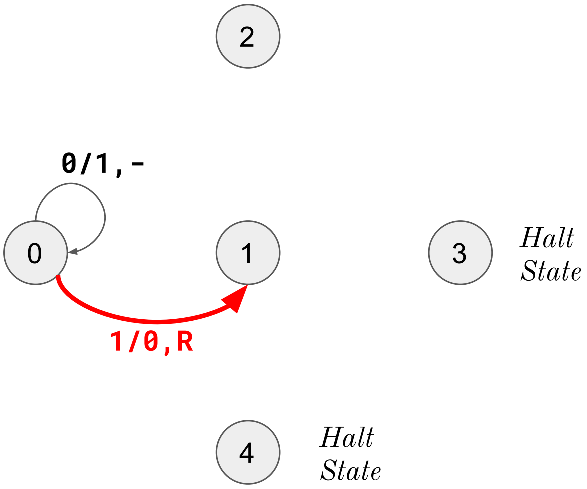
```
if read 0:
    write 1
    no move
    goto 0
```

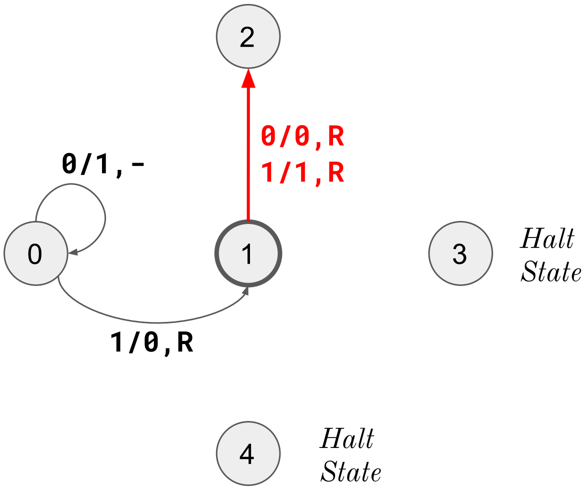
```
if read 1:
    write 0
    move RIGHT
    goto 1
```



```
if read 0:
    write 1
    no move
    goto 0
```

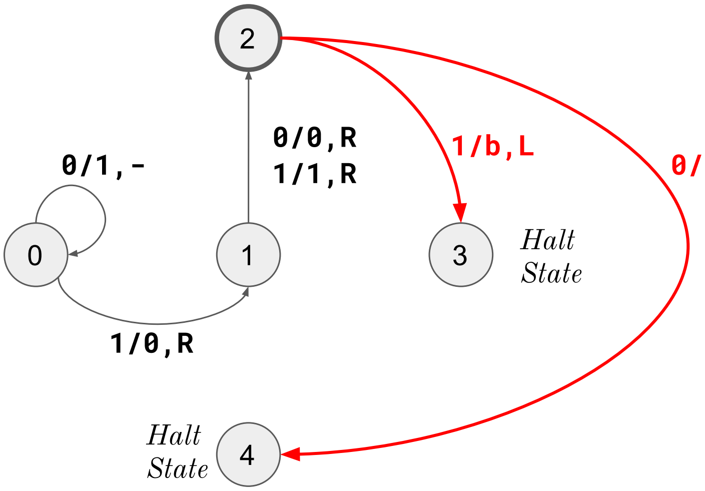
```
if read 1:
    write 0
    move RIGHT
    goto 1
```





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
```



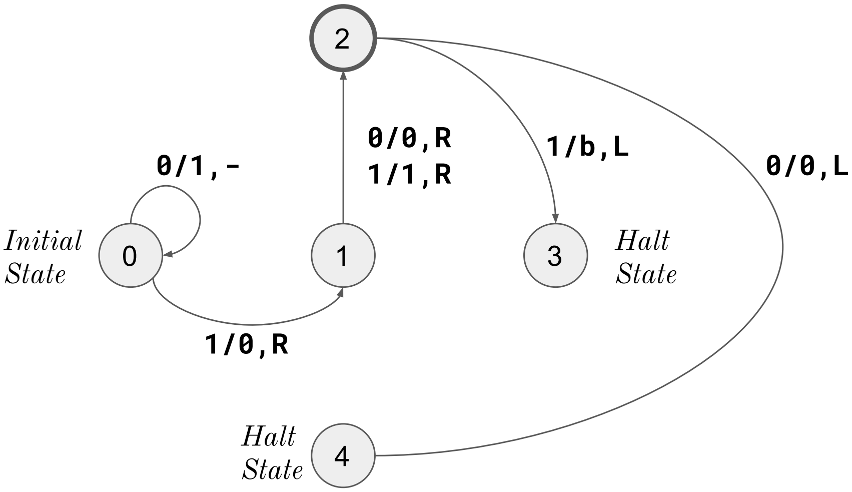
What is the instructions for state 2?

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?


2
0/0,R
1/1,R
1/b,L
0/0,L
0/1,-
Initial State
0
1
3
Halt State
1/0,R
Halt State
4