

## **Ejercicios sobre máquinas de Turing**

Julian Esteban Ballesteros Ortiz

Programa de ingeniería de sistemas y computación, Universidad de Cundinamarca

Lenguajes y Autómatas 801 ISC TRAN

Docente

Ing. Fabio Alejandro Sastoque Rincón

Miércoles 1 de octubre de 2025

## Problema 1



LOAD MACHINE

REVERT TO DIAGRAM

```

1 input: "10110" # código para probar otras entradas
2 blank: "."
3 start state: q0
4 table:
5   q0:
6     0: (write: 1, R)
7     1: (write: 0, R)
8     .: (blank, done)
9 done:
10

```



```

2025 - 2 > Lenguajes y automatas > turing1.py > ...
1 class TuringMachineComplemento:
2     def __init__(self, tape):
3         self.tape = list(tape + "_")
4         self.head = 0
5         self.state = "q0"
6         self.final_states = {"halt"}
7         self.transition_function = {
8             ("q0", "0"): ("1", "R", "q0"),
9             ("q0", "1"): ("0", "R", "q0"),
10            ("q0", "."): (".", "R", "halt")
11        }
12
13    def step(self):
14        symbol = self.tape[self.head]
15        key = (self.state, symbol)
16
17        if key not in self.transition_function:
18            self.state = "halt"
19            return False
20
21        new_symbol, direction, new_state = self.transition_function[key]
22        self.tape[self.head] = new_symbol
23
24        if direction == "R":
25            self.head += 1
26        elif direction == "L":

```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

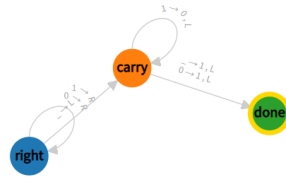
PORTS

```

PS C:\Users\Julia\Documents\Universidad> & C:/Users/Julia/AppData/Local/Programs/Python/Pyt
py"
Entrada: 10110
Complemento: 01001

```

## Problema 2



```

1 input: '111' # prueba '1011', '011' ...
2 blank: '-'
3 start state: right
4 table:
5- right:
6   1: R
7   0: R
8   -: (L: carry)
9- carry:
10  1: (write: 0, L)
11  0: (write: 1, L: done)
12  -: (write: 1, L: done)
13 done:
14

```



```

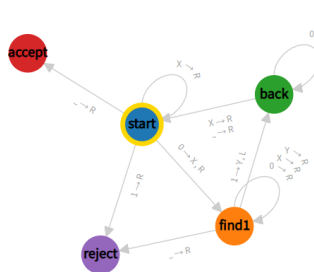
2025 - 2 > Lenguajes y automatas > turing2.py > ...
1 class TuringMachineSucesor:
2     def __init__(self, tape):
3
4
5
6
7
8
9         ("q0", "1"): ("0", "R", "q0"), # 1 + carry -> 0 y seguimos
10        ("q0", "0"): ("1", "R", "halt"), # 0 + 1 -> 1 y terminamos
11        ("q0", "-"): ("1", "R", "halt") # overflow -> agregamos 1
12    }
13
14    def step(self):
15        symbol = self.tape[self.head]
16        key = (self.state, symbol)
17
18        if key not in self.transition_function:
19            self.state = "halt"
20            return False
21
22        new_symbol, direction, new_state = self.transition_function[key]
23        self.tape[self.head] = new_symbol
24
25        if direction == "R":
26            self.head += 1
27        elif direction == "L":
28            self.head = max(0, self.head - 1)
29
30        self.state = new_state
31        return True
32
33    def run(self):
34        while self.state not in self.final_states:
35            self.step()
36        result = "".join(self.tape).strip("-")
37        return result[::-1] # devolvemos el número en orden normal
38
39
40 # PRUEBA
41 binario = "111"
42 tm = TuringMachineSucesor(binario)
43 print("Entrada:", binario)
44 print("Sucesor:", tm.run())
45

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\Julia\Documents\Universidad> & C:/Users/Julia/AppData/Local/Programs/Python/Python3.10.0/Python.exe C:\Users\Julia\Documents\Universidad\turing2.py  
Entrada: 111  
Sucesor: 1000  
PS C:\Users\Julia\Documents\Universidad> |

### Problema 3



```

1 # Acepta cadenas de la forma 0^n
2 input: "0011"
3 blank: "."
4 start state: start
5 accept states: [accept]
6 reject states: [reject]
7
8 table:
9   start:
10    0: {write: X, R: find1}
11    X: {R: start}
12    ".": {R: accept}
13    1: {R: reject}
14
15   find1:
16    0: {R: find1}
17    X: {R: find1}
18    Y: {R: find1}
19    1: {write: Y, L: back}
20    ".": {R: reject}
21
22   back:
23    0: {L: back}
24    1: {L: back}
25    Y: {L: back}
26    X: {R: find1}
27    ".": {L: reject}
28
29   accept:
30   reject:

```

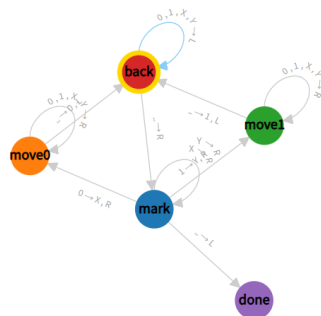


```

2025 - 2 > Lenguajes y automatas > turing3.py > ...
1 class TuringMachine0n1n:
2     def __init__(self, tape):
13         ("q1", "_"): ("_", "R", "reject"),
14
15         ("q2", "0"): ("0", "L", "q2"),
16         ("q2", "_"): ("_", "R", "q0")
17     }
18
19     def step(self):
20         symbol = self.tape[self.head]
21         key = (self.state, symbol)
22
23         if key not in self.transition_function:
24             self.state = "reject"
25             return False
26
27         new_symbol, direction, new_state = self.transition_function[key]
28         self.tape[self.head] = new_symbol
29
30         if direction == "R":
31             self.head += 1
32         elif direction == "L":
33             self.head = max(0, self.head - 1)
34
35         self.state = new_state
36         return True
37
38     def run(self):
39         while self.state not in self.final_states:
40             self.step()
41         return self.state == "accept"
42
43
44 # PRUEBA
45 cadena = "0011"
46 tm = TuringMachine0n1n(cadena)
47 print("Entrada:", cadena)
48 print("¿Pertenece a {0^n1^n}?:", tm.run())
49
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Julia\Documents\Universidad> & C:/Users/Julia/AppData/Local/Programs/Python/P
Entrada: 0011
¿Pertenece a {0^n1^n}?: False

```

## Problema 4



```

1 # Reversa de una palabra binaria
2 input: '101100'
3 blank: ' '
4 start state: mark
5 table:
6   mark:
7     0: {write: X, R: move0}
8     1: {write: Y, R: move1}
9     X: R
10    Y: R
11    '': {L: done}
12
13   move0:
14     [0,1,X,Y]: R
15     '': {write: 0, L: back}
16
17   move1:
18     [0,1,X,Y]: R
19     '': {write: 1, L: back}
20
21   back:
22     [0,1,X,Y]: L
23     '': {R: mark}
24
25   done:

```

			Y	X	Y	Y	X	X	1	0	1	1	0	0			
--	--	--	---	---	---	---	---	---	---	---	---	---	---	---	--	--	--

2025 - 2 > Lenguajes y automatas > turing4.py > ...

```

1 class TuringMachineReverso:
2     def __init__(self, tape):
3         self.tape = tape
4
5     def run(self):
6         return self.tape[::-1]
7
8
9 # PRUEBA
10 cadena = "101100"
11 tm = TuringMachineReverso(cadena)
12 print("Entrada:", cadena)
13 print("Reverso:", tm.run())
14

```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

```

PS C:\Users\Julia\Documents\Universidad> &
/Julia/Documents/Universidad/2025 - 2/Lengu
Entrada: 101100
Reverso: 001101

```

