



UNIVERSIDAD AUTÓNOMA DE  
**CHIHUAHUA**

UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA  
Facultad de Ingeniería



Ingeniería en Ciencias de la Computación

## **TEORÍA DE LA COMPUTACIÓN**

### **Programa Expresión Regular**

*Trabajo de:* ADRIAN A. GONZÁLEZ DOMÍNGUEZ [359834]  
*Asesor:* MARIO ANDRES CUEVAS GUTIERREZ

*16 de septiembre de 2024*

## Programa expresión regular

Adrián A. González Domínguez 359834

Programa en Python

#Automata A

def automaton():

states=[

[0,1,3], #0

[0,2,3], #1

[2,2,3], #2

[3,3,3], #3

]

finals=[2]

alphabet=['a','b']

state=0

result=[]

while True:

c=input\_file.read(1)

if not c or c=='\n':

break

result.append(c)

column=choose\_column(

len(alphabet), alphabet, c

)

state=states[state][column]

return is\_final(

len(finals), finals, state

), ''.join(result)

#Automata B

def automaton():

states=[

[2,1,4], #0

[1,1,4], #1

[2,3,4], #2

[2,3,4], #3

[4,4,4], #4

]

finals=[1,3]

alphabet=['a','b']

state=0

result=[]

while True:

c=input\_file.read(1)

if not c or c=='\n':

break

result.append(c)

column=choose\_column(

len(alphabet), alphabet, c

)

state=states[state][column]

return is\_final(

len(finals), finals, state

), ''.join(result)



```
def choose_column(alphabet_length, alphabet, c):
```

```
    for i in range(alphabet_length):
```

```
        if alphabet[i] == c:
```

```
            return i
```

```
    return alphabet_length
```

```
def is_final(finals_length, finals, state):
```

```
    return state in finals
```

```
if __name__ == '__main__':
```

```
    input_file = open(0)
```

```
    while True:
```

```
        valid, result = eat_maton()
```

```
        if result == '':
```

```
            break
```

```
        if valid:
```

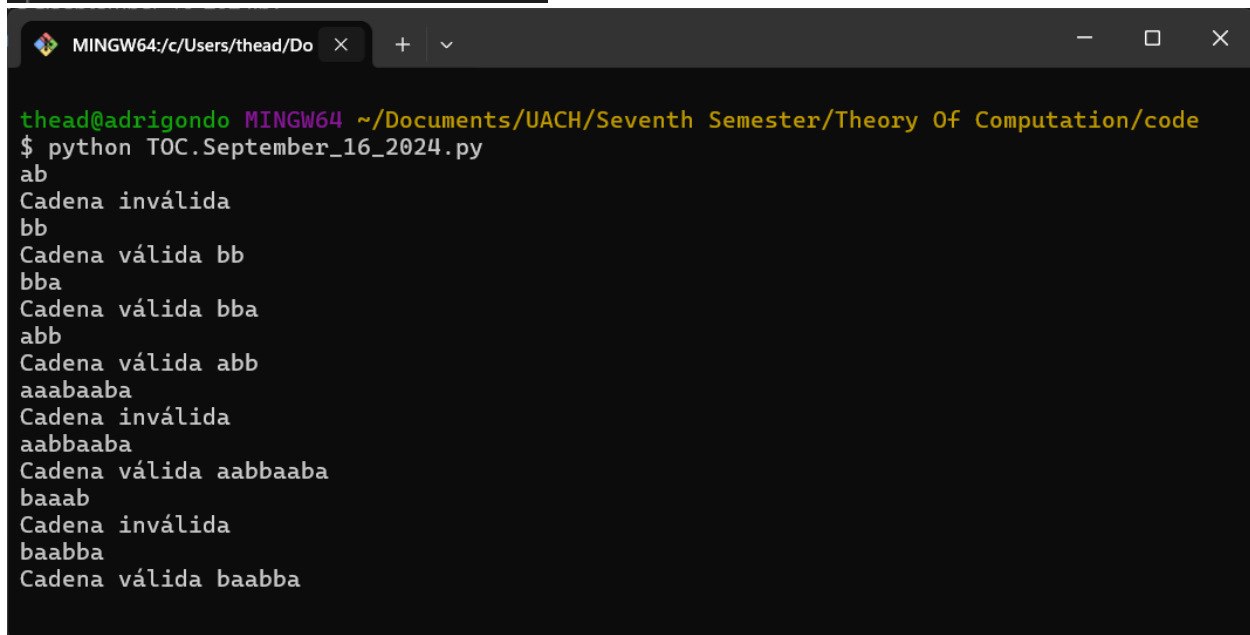
```
            print(f"Cadena valida: {result}")
```

```
        else:
```

```
            print(f"Cadena inválida")
```

Para el automata A

```
def automaton():  
    states = [  
        [0, 1, 3], # 0  
        [0, 2, 3], # 1  
        [2, 2, 3], # 2  
        [3, 3, 3], # 3  
    ]  
    finals = [2]
```

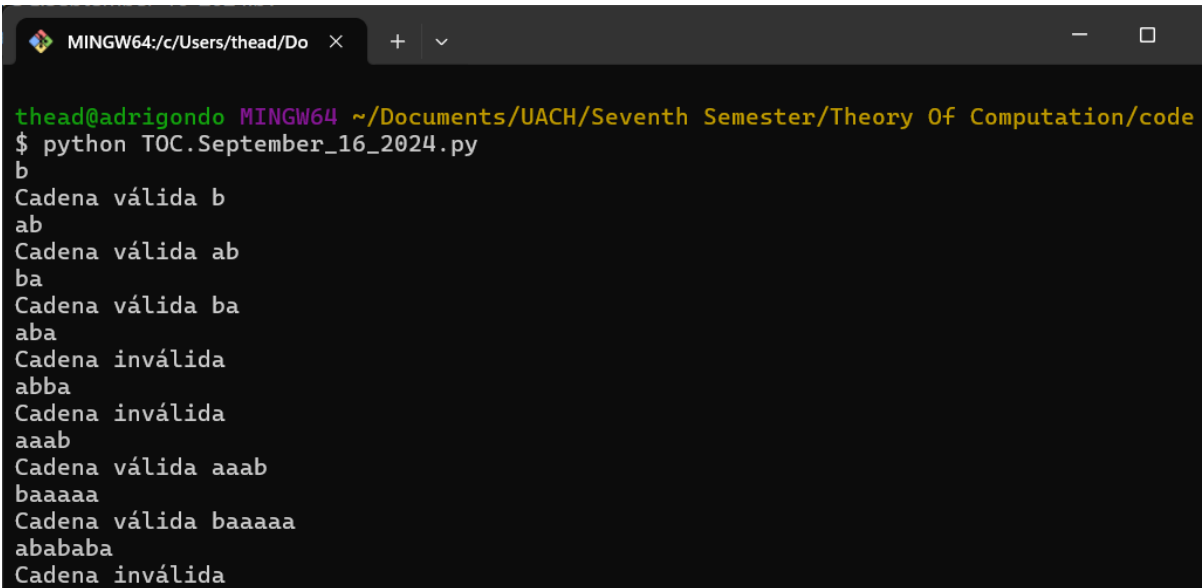


A terminal window titled 'MINGW64: c/Users/thead/Do' showing the execution of a Python script. The prompt is 'thead@adrigondo MINGW64 ~/Documents/UACH/Seventh Semester/Theory Of Computation/code'. The command '\$ python TOC.September\_16\_2024.py' has been executed, resulting in a series of input strings and their corresponding validity status: 'ab' is 'Cadena inválida', 'bb' is 'Cadena válida bb', 'bba' is 'Cadena válida bba', 'abb' is 'Cadena válida abb', 'aaabaaba' is 'Cadena inválida', 'aabbaaba' is 'Cadena válida aabbaaba', 'baaab' is 'Cadena inválida', 'baabba' is 'Cadena válida baabba'.

```
thead@adrigondo MINGW64 ~/Documents/UACH/Seventh Semester/Theory Of Computation/code  
$ python TOC.September_16_2024.py  
ab  
Cadena inválida  
bb  
Cadena válida bb  
bba  
Cadena válida bba  
abb  
Cadena válida abb  
aaabaaba  
Cadena inválida  
aabbaaba  
Cadena válida aabbaaba  
baaab  
Cadena inválida  
baabba  
Cadena válida baabba
```

Para el automata B

```
def automaton():  
    states = [  
        [2, 1, 4], # 0  
        [1, 1, 4], # 1  
        [2, 3, 4], # 2  
        [2, 3, 4], # 3  
        [4, 4, 4], # 4  
    ]  
    finals = [1, 3]
```



A terminal window titled 'MINGW64:/c/Users/thead/Do' showing the execution of a Python script. The prompt is 'thead@adrigondo MINGW64 ~/Documents/UACH/Seventh Semester/Theory Of Computation/code'. The command executed is '\$ python TOC.September\_16\_2024.py'. The output shows a series of inputs and their corresponding validity status: 'b' is 'Cadena válida', 'ab' is 'Cadena válida', 'ba' is 'Cadena válida', 'aba' is 'Cadena inválida', 'abba' is 'Cadena inválida', 'aaab' is 'Cadena válida', 'baaaaa' is 'Cadena válida', and 'abababa' is 'Cadena inválida'.

```
thead@adrigondo MINGW64 ~/Documents/UACH/Seventh Semester/Theory Of Computation/code  
$ python TOC.September_16_2024.py  
b  
Cadena válida b  
ab  
Cadena válida ab  
ba  
Cadena válida ba  
aba  
Cadena inválida  
abba  
Cadena inválida  
aaab  
Cadena válida aaab  
baaaaa  
Cadena válida baaaaa  
abababa  
Cadena inválida
```