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

Teoría de la Computación — Lunes 16/Septiembre/2024

Programa expresión regular

Adrian 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.apend(c)
    column=choose_column(
      len(alphabet),alphabet,c
    )
    state=states[state][column]
  return is_finall
    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.apend(c)
    column=choose_column(
      len(alphabet), alphabet, c
    )
    state=states[state][column]
  retorn is_final(
    len(finals), finals, state
  ),''.join(resault)
```

```
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 = automaton()
    if result != '':
      break
  if valid:
    print(f"Cadena valida: {result}")
  else:
    print(f"Cadena invalida")
```

Para el automata A

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

```
MINGW64:/c/Users/thead/Do

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

```python
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]
```