

Instituto Politécnico Nacional

Escuela Superior de Cómputo



Programa 9: Máquina De Turing

Autor: Colín Ramiro Joel

Materia: Teoría de la Computación

Grupo: 4CM2

Profesor: Juarez Martínez Genaro

Fecha de entrega: **29 de Diciembre 2021**

Introducción

La Máquina de Turing es talvez uno de los temas más interesantes y fascinantes de explicar en las ciencias de la computación. No es nada más que un dispositivo el cual manipula símbolos sobre una tira de cinta de acuerdo con una tabla de reglas. A pesar de su simplicidad, una máquina de Turing puede ser adaptada para simular la lógica de cualquier algoritmo de computador y es muy útil en la explicación de las funciones de una CPU dentro de una computadora. Esta máquina fue concebida por el matemático por el cual adoptó su nombre **ALAN TURING**, originalmente estaba destinada en el estudio la cuestión de otro matemático llamado David Hilbert, sobre si las matemáticas son decidibles, es decir, si existe un método definido que pueda aplicarse a cualquier sentencia matemática y que pueda decir con certeza si esa sentencia es cierta o no. Turing ideó este modelo formal de computadora y demostró que existían problemas que una máquina no podía resolver. Ahora bien la máquina de Turing que sea capaz de simular cualquier otra máquina de Turing es llamada una máquina universal de Turing (UTM).

La importancia de la máquina de Turing en la historia de la computación es en partida doble:

1. En primer lugar, la máquina de Turing fue uno de los primeros modelos teóricos para las computadoras en el año de 1936.
2. Y en segundo, estudiando sus propiedades abstractas, la máquina de Turing ha servido de base para mucho desarrollo teórico en las ciencias de la computación y en la teoría de la complejidad.

Instrucciones

Programar la máquina de la tabla 1 del artículo de arXiv: "What can we learn from universal Turing machines?".

1. La máquina se tiene que animar para cadenas pequeñas (≤ 10 caracteres).
2. Puede recibir la cadena por parte del usuario o aleatoriamente.
3. Mandar la salida a un archivo de texto que muestre las descripciones instantáneas por renglón en cada iteración.

Desarrollo

Como bien se especificó en la parte de la introducción se programó la máquina de la tabla 1 (**table 1**) del artículo antes mencionado. Para esto recurrí de un análisis profundo a la tabla, para poder ver la máquina de Turing como un autómata y así poder tener más claro el como realizarlo.

Posterior a este análisis se comenzó a codificar la máquina. Para comenzar es importante hacer notar que al igual que algunos otros programas anteriores,

cuenta con un menú principal en el cual el usuario tiene el poder de elegir la forma de introducir la cadena, ya sea de forma manual o automática. Además de la ya conocida opción para salir de dicho programa. Una vez realizado el menú, implementé una función llamada **maqTuring(opcion)**, la cual recibe como argumento la opción seleccionada por el usuario, para su evaluación. Esta función lo que realiza es que evalúa la cadena en función a las reglas establecidas en la tabla misma. Este programa genera un archivo de texto llamado **SalidaP9.txt**, en el cual, se encuentran las Descripciones Instantáneas (IDs) de cada iteración para que se pueda observar con más detalle las transiciones que se están realizando. Estas IDs son una tripleta con la estructura que menciona el autor del PDF sugerido el cual es el siguiente: **(A, M, s)**.

Donde:

- **A** = Símbolo o carácter que se sobrescribe en la casilla
- **M** = Movimiento que se realiza en la cinta (R=Derecha, L=Izquierda, Z=No hay movimiento)
- **s** = Estado siguiente

Algo importante a notar es que si la longitud de la cadena es menor o igual a los 10 caracteres se debía animar la máquina de Turing. En este arpatado no logre animarlo del todo, solo despliega un esquema sencillo de como sería la máquina estática. En la sección **Capturas del Funcionamiento** se verá más claro este esquema. Finalmente, tanto en la consola como en el archivo de texto, se especifica si la cadena es aceptada o rechazada.

Table 1 *Table of the Turing machine for the courteous addition.*

	-	*		a	X
1		X R 2			
2		R 3	R		
3		X L 4	R		
4		L	a R 5		R 7
5	L 6	R	R		R
6		L	L	L 4	L
7		R 8	R		
8	* L 9		R		* R
9		L	L		* !

Capturas del Funcionamiento

En esta sección al igual que en los anteriores programas, se encuentran las capturas de pantalla del funcionamiento del programa.

1. Cadena Manual

```
C:\Users\joelc\OneDrive\Documentos\Joel\4to\TC\Programas\Programa9-Máquina de Turing>python Programa9_TC.py

*****Menu Principal*****

****Digite la opción****

1.- Cadena Manual
2.- Cadena Automática
3.- Salir
1

Ingrese la cadena deseada (Solo puede contener los símbolos: '-', '*', '|', 'a' y 'X'):
-
```

```
1
Ingrese la cadena deseada (Solo puede contener los símbolos: '-', '*', '|', 'a' y 'X'):
***|||aa*XXX

La cadena a evaluar es: ['*', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
Caracter analizado: *
El caracter sobre escrito es: X
(X,R,q1)
['*', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
Caracter analizado: *
El caracter sobre escrito es: *
(*,R,q2)
['*', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
Caracter analizado: *
El caracter sobre escrito es: X
(X,L,q3)
['*', '*', 'X', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
Caracter analizado: *
El caracter sobre escrito es: *
(*,L,q3)
['*', '*', 'X', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']

Salida: Blo de notas
Ayuda: Edición, formato, Imprimir, Ayuda
*En este archivo se encuentran las descripciones instantáneas([Blo] de la Máquina de Turing.*
La cadena a evaluar es: ['*', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
Caracter analizado: *
El caracter sobre escrito es: X
La cadena sobrescrita es: ['X', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID1.- (*,R,q1)
Caracter analizado: *
El caracter sobre escrito es: *
La cadena sobrescrita es: ['X', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID2.- (*,R,q2)
Caracter analizado: *
El caracter sobre escrito es: X
La cadena sobrescrita es: ['X', '*', 'X', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID3.- (X,L,q3)
Caracter analizado: *
El caracter sobre escrito es: *
La cadena sobrescrita es: ['X', '*', 'X', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID4.- (*,L,q3)
Caracter analizado: X
El caracter sobre escrito es: X
La cadena sobrescrita es: ['X', '*', 'X', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID5.- (X,R,q8)
Caracter analizado: *
El caracter sobre escrito es: *
La cadena sobrescrita es: ['X', '*', 'X', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID6.- (*,R,q7)
Caracter analizado: X
El caracter sobre escrito es: X
La cadena sobrescrita es: ['X', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID7.- (*,R,q7)
Caracter analizado: -
El caracter sobre escrito es: *
La cadena sobrescrita es: ['X', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID8.- (*,L,q8)
Caracter analizado: *
El caracter sobre escrito es: *
La cadena sobrescrita es: ['X', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID9.- (*,L,q8)
Caracter analizado: *
El caracter sobre escrito es: *
La cadena sobrescrita es: ['X', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID10.- (*,L,q8)
Caracter analizado: X
Cadena Aceptada
La cadena final es: ['*', '*', '*', '-', '|', '|', '|', 'a', 'a', '*', 'X', 'X', 'X']
ID11.- (*,Z,F)
```

2. Cadena Automática

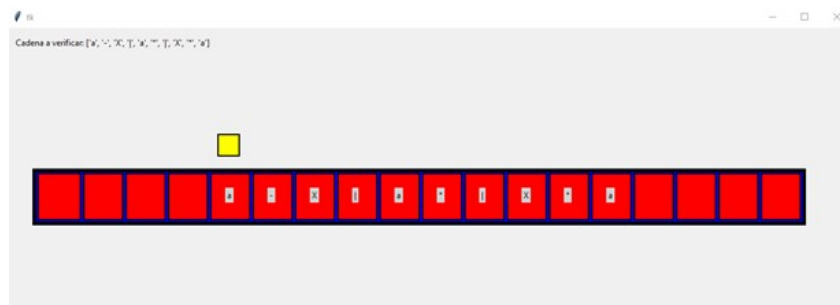
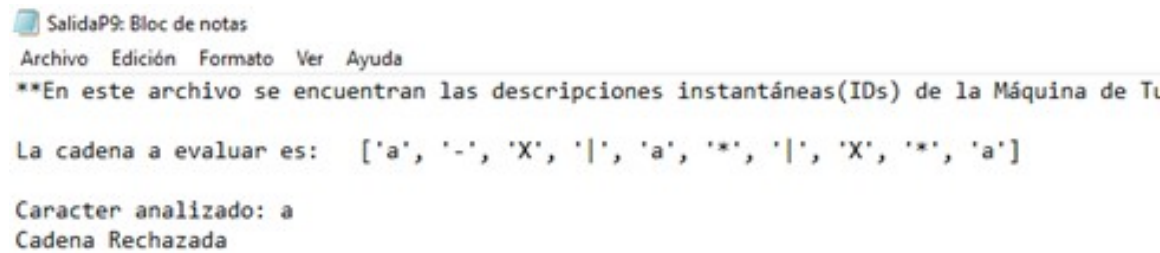
```
C:\Users\joelc\OneDrive\Documentos\Joel\4to\TC\Programas\Programa9-Máquina de Turing>python Programa9_TC.py

*****Menu Principal*****

****Dígitela opción****

1.- Cadena Manual
2.- Cadena Automática
3.- Salir
2

La cadena a evaluar es: ['a', '-', 'X', '|', 'a', '*', '|', 'X', '*', 'a']
Caracter analizado: a
Cadena Rechazada
```



Código

```
# Programa 9. Máquina de Turing
# Nombre: Colín Ramiro Joel
# Profesor: Juárez Martínez Genaro
# Grupo: 4CM2
# Materia: Teoría Computacional
import sys
import time
import random
from tkinter import *
from random import randint

def maqTuring(opc):
    archivo = open("SalidaP9.txt", "w")
    archivo.write("**En este archivo se encuentran las descripciones instantáneas(IDs) de la Máquina de Turing.**\n\n")
```

```

alfabeto = ["-", "*", "|", "a", "X"]
if(opc == 1):
    cadena = input("Ingrese la cadena deseada (Solo puede
                    contener los s mbolos: '-', '*',
                    '|', 'a' y 'X'): \n")

elif(opc == 2):
    tamCad = randint(0,100)
    cadena = ""
    for i in range(0, tamCad):
        cadena = cadena + random.choice(alfabeto)
    cad = []
    for elem in cadena:
        cad.append(elem)
    print("\n")
    if(len(cad) <= 10):
        ventana = Tk()
        canva = Canvas(ventana , width=1200, height =400)
        ventana.geometry("1200x400")
        canva.pack()
        canva.create_rectangle(39, 200, 1130, 275, width=5, fill='
                                blue')

        xIni = 45
        xFin = 105
        yIni = 205
        yFin = 270
        mens = Label(ventana, text = "Cadena a verificar: " + str(
                                cad)).place(x=10,y=10)

        xCabezaIni = 300
        xCabezaFin = 330
        canva.create_rectangle(xCabezaIni, 150, xCabezaFin, 180,
                                width=2, fill='yellow')

        caract = ""
        for i in range(0,18):
            canva.create_rectangle(xIni, yIni, xFin, yFin, width=2,
                                    fill='red')

            xIni = xFin + 5
            xFin = xFin + 60
        posxCad = 310
        posyCad = 225
        for car in cad:
            caract = caract + car
            caracterAnim = Label(ventana, text=car, bg="light gray").
                place(x=posxCad,y=posyCad)

            posxCad = posxCad + 60
    estado = "q0"
    i = 0
    cont = 1
    print("La cadena a evaluar es: " + str(cad))
    archivo.write("La cadena a evaluar es: " + str(cad) + "\n\n"
                ")
    try:
        while(estado != "q9"):
            pos = 0
            posaux = 0
            for elem in cad:
                if(pos == i):
                    pos = pos+1

```

```

if(i == -1):
    cad.insert(0, "")
    i = 0
if(estado == "q0"): #Estado q0 = 1
    if(cad[i] == "*"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q1"
        cad[i] = "X"
        i = i+1
        print("El caracter sobre escrito es: X\\n")
        archivo.write("El caracter sobre escrito es: X\\n")
        print("(X,R,"+estado+")\\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\\n")
        archivo.write("ID"+ str(cont) + ".- (X,R,"+estado+")\\n
            \\n")

        cont = cont+1
    else:
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q0"
        cad[i] = i
        print("(" + cad[i] + ",R,"+estado+")\\n")
        print("Cadena Rechazada")
        break

if(estado == "q1"): #Estado q1 = 2
    if(cad[i] == "*"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q2"
        cad[i] = "*"
        i = i+1
        print("El caracter sobre escrito es: *\\n")
        archivo.write("El caracter sobre escrito es: *\\n")
        print("(*,R,"+estado+")\\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\\n")
        archivo.write("ID"+ str(cont) + ".- (*,R,"+estado+")\\n
            \\n")

        cont = cont+1
    elif(cad[i] == "|"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q1"
        cad[i] = "|"
        i = i+1
        print("El caracter sobre escrito es: |\\n")
        archivo.write("El caracter sobre escrito es: |\\n")
        print("(|,R,"+estado+")\\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\\n")

```

```

        archivo.write("ID"+ str(cont) +".- (|,R,"+estado+")\n
                        \n")

        cont = cont+1
    else:
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+" \n")
        estado = "q1"
        cad[i] = i
        print("(" +cad[i]+",Z,"+estado+")\n")
        print("Cadena Rechazada")
        break
if(estado == "q2"): #Estado q2 = 3
    if(cad[i] == "*"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+" \n")
        estado = "q3"
        cad[i] = "X"
        i = i-1
        print("El caracter sobre escrito es: X\n")
        archivo.write("El caracter sobre escrito es: X\n")
        print("(X,L,"+estado+")\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
                        + " \n")
        archivo.write("ID"+ str(cont) +".- (X,L,"+estado+")\n
                        \n")

        cont = cont+1
    elif(cad[i] == "|"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+" \n")
        estado = "q2"
        cad[i] = "|"
        i = i+1
        print("El caracter sobre escrito es: |\n")
        archivo.write("El caracter sobre escrito es: |\n")
        print("(|,R,"+estado+")\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
                        + " \n")
        archivo.write("ID"+ str(cont) +".- (|,R,"+estado+")\n
                        \n")

        cont = cont+1
    else:
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+" \n")
        estado = "q2"
        cad[i] = i
        print("(" +cad[i]+",Z,"+estado+")\n")
        print("Cadena Rechazada")
        break

if(estado == "q3"): #Estado q3 = 4
    if(cad[i] == "*"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+" \n")
        estado = "q3"
        cad[i] = "*"

```



```

i = i-1
print("El caracter sobre escrito es: *\n")
archivo.write("El caracter sobre escrito es: *\n")
print("(*,L,"+estado+")\n")
print(cad)
archivo.write("La cadena sobrescrita es: " + str(cad)
              + "\n")
archivo.write("ID"+ str(cont) + ".- (*,L,"+estado+")\n
              \n")

cont = cont+1
elif(cad[i] == "|"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+\n")
    estado = "q4"
    cad[i] = "a"
    i = i+1
    print("El caracter sobre escrito es: a\n")
    archivo.write("El caracter sobre escrito es: a\n")
    print("(a,R,"+estado+")\n")
    print(cad)
    archivo.write("La cadena sobrescrita es: " + str(cad)
                  + "\n")
    archivo.write("ID"+ str(cont) + ".- (a,R,"+estado+")\n
                  \n")

    cont = cont+1
elif(cad[i] == "X"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+\n")
    estado = "q6"
    cad[i] = "X"
    i = i+1
    print("El caracter sobre escrito es: X\n")
    archivo.write("El caracter sobre escrito es: X\n")
    print("(X,R,"+estado+")\n")
    print(cad)
    archivo.write("La cadena sobrescrita es: " + str(cad)
                  + "\n")
    archivo.write("ID"+ str(cont) + ".- (X,R,"+estado+")\n
                  \n")

    cont = cont+1
else:
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+\n")
    estado = "q3"
    cad[i] = i
    print("(" + cad[i] + ",Z,"+estado+")\n")
    print("Cadena Rechazada")
    break

if(estado == "q4"): #Estado q4 = 5
    if(cad[i] == "-"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+\n")
        estado = "q5"
        cad[i] = "|"
        i = i-1
        print("El caracter sobre escrito es: |\n")

```

```

archivo.write("El caracter sobre escrito es: |\n")
print("(|,L,"+estado+")\n")
print(cad)
archivo.write("La cadena sobrescrita es: " + str(cad)
              + "\n")
archivo.write("ID"+ str(cont) + ".- (|,L,"+estado+")\n
              \n")

cont = cont+1
elif(cad[i] == "*"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+\n")
    estado = "q4"
    cad[i] = "*"
    i = i+1
    print("El caracter sobre escrito es: *\n")
    archivo.write("El caracter sobre escrito es: *\n")
    print("(*,R,"+estado+")\n")
    print(cad)
    archivo.write("La cadena sobrescrita es: " + str(cad)
                  + "\n")
    archivo.write("ID"+ str(cont) + ".- (*,R,"+estado+)\n
                  \n")

cont = cont+1
elif(cad[i] == "|"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+\n")
    estado = "q4"
    cad[i] = "|"
    i = i+1
    print("El caracter sobre escrito es: |\n")
    archivo.write("El caracter sobre escrito es: |\n")
    print("(|,R,"+estado+")\n")
    print(cad)
    archivo.write("La cadena sobrescrita es: " + str(cad)
                  + "\n")
    archivo.write("ID"+ str(cont) + ".- (|,R,"+estado+)\n
                  \n")

cont = cont+1
elif(cad[i] == "X"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+\n")
    estado = "q4"
    cad[i] = "X"
    i = i+1
    print("El caracter sobre escrito es: X\n")
    archivo.write("El caracter sobre escrito es: X\n")
    print("(X,R,"+estado+")\n")
    print(cad)
    archivo.write("La cadena sobrescrita es: " + str(cad)
                  + "\n")
    archivo.write("ID"+ str(cont) + ".- (X,R,"+estado+)\n
                  \n")

cont = cont+1
else:
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+\n")
    estado = "q4"

```

```

cad[i] = i
print("(" + cad[i] + ",Z," + estado + ")\n")
print("Cadena Rechazada")
break

if(estado == "q5"): #Estado q5 = 6
    if(cad[i] == "*"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+\n")
        estado = "q5"
        cad[i] = "*"
        i = i-1
        print("El caracter sobre escrito es: *\n")
        archivo.write("El caracter sobre escrito es: *\n")
        print("(*,L," + estado + ")\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\n")
        archivo.write("ID"+ str(cont) + ".- (*,L," + estado + ")\n
            \n")

        cont = cont+1
    elif(cad[i] == "|"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+\n")
        estado = "q5"
        cad[i] = "|"
        i = i-1
        print("El caracter sobre escrito es: |\n")
        archivo.write("El caracter sobre escrito es: |\n")
        print("(|,L," + estado + ")\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\n")
        archivo.write("ID"+ str(cont) + ".- (|,L," + estado + ")\n
            \n")

        cont = cont+1
    elif(cad[i] == "a"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+\n")
        estado = "q3"
        cad[i] = "|"
        i = i-1
        print("El caracter sobre escrito es: |\n")
        archivo.write("El caracter sobre escrito es: |\n")
        print("(|,L," + estado + ")\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\n")
        archivo.write("ID"+ str(cont) + ".- (|,L," + estado + ")\n
            \n")

        cont = cont+1
    elif(cad[i] == "X"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+\n")
        estado = "q5"
        cad[i] = "X"
        i = i-1

```

```

print("El caracter sobre escrito es: X\n")
archivo.write("El caracter sobre escrito es: X\n")
print("(X,L,"+estado+")\n")
print(cad)
archivo.write("La cadena sobrescrita es: " + str(cad)
              + "\n")
archivo.write("ID"+ str(cont) + ".- (X,L,"+estado+")\n"
              \n")
cont = cont+1
else:
print("Caracter analizado: " + cad[i] + "\n")
archivo.write("Caracter analizado: "+cad[i]+\n")
estado = "q5"
cad[i] = i
print("(" + cad[i] + ",Z," + estado + ")\n")
print("Cadena Rechazada")
break

if(estado == "q6"): #Estado q6 = 7
if(cad[i] == "*"):
print("Caracter analizado: " + cad[i] + "\n")
archivo.write("Caracter analizado: "+cad[i]+\n")
estado = "q7"
cad[i] = "*"
i = i+1
print("El caracter sobre escrito es: *\n")
archivo.write("El caracter sobre escrito es: *\n")
print("(*,R,"+estado+")\n")
print(cad)
archivo.write("La cadena sobrescrita es: " + str(cad)
              + "\n")
archivo.write("ID"+ str(cont) + ".- (*,R,"+estado+)\n"
              \n")

cont = cont+1
elif(cad[i] == "|"):
print("Caracter analizado: " + cad[i] + "\n")
archivo.write("Caracter analizado: "+cad[i]+\n")
estado = "q6"
cad[i] = "|"
i = i+1
print("El caracter sobre escrito es: |\n")
archivo.write("El caracter sobre escrito es: |\n")
print("(|,R,"+estado+")\n")
print(cad)
archivo.write("La cadena sobrescrita es: " + str(cad)
              + "\n")
archivo.write("ID"+ str(cont) + ".- (|,R,"+estado+)\n"
              \n")

cont = cont+1
else:
print("Caracter analizado: " + cad[i] + "\n")
archivo.write("Caracter analizado: "+cad[i]+\n")
estado = "q6"
cad[i] = i
print("(" + cad[i] + ",Z," + estado + ")\n")
print("Cadena Rechazada")
break

```

```

if(estado == "q7"): #Estado q7 = 8
    if(cad[i] == "-"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q8"
        cad[i] = "*"
        i = i-1
        print("El caracter sobre escrito es: *\n")
        archivo.write("El caracter sobre escrito es: *\n")
        print("(*,L,"+estado+")\\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\\n")
        archivo.write("ID"+ str(cont) + ".- (*,L,"+estado+")\\n
            \\n")

        cont = cont+1
    elif(cad[i] == "|"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q7"
        cad[i] = "|"
        i = i+1
        print("El caracter sobre escrito es: |\n")
        archivo.write("El caracter sobre escrito es: |\n")
        print("(|,R,"+estado+")\\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\\n")
        archivo.write("ID"+ str(cont) + ".- (|,R,"+estado+")\\n
            \\n")

        cont = cont+1
    elif(cad[i] == "X"):
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q7"
        cad[i] = "*"
        i = i+1
        print("El caracter sobre escrito es: X\\n")
        archivo.write("El caracter sobre escrito es: X\\n")
        print("(*,R,"+estado+")\\n")
        print(cad)
        archivo.write("La cadena sobrescrita es: " + str(cad)
            + "\\n")
        archivo.write("ID"+ str(cont) + ".- (*,R,"+estado+")\\n
            \\n")

        cont = cont+1
    else:
        print("Caracter analizado: " + cad[i] + "\n")
        archivo.write("Caracter analizado: "+cad[i]+"\\n")
        estado = "q7"
        cad[i] = i
        print((" "+cad[i]+",Z,"+estado+")\\n")
        print("Cadena Rechazada")
        break

if(estado == "q8"): #Estado q8 = 9

```

```

if(cad[i] == "*"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+"\\n")
    estado = "q8"
    cad[i] = "*"
    i = i-1
    print("El caracter sobre escrito es: *\n")
    archivo.write("El caracter sobre escrito es: *\n")
    print("(*,L,"+estado+")\\n")
    print(cad)
    archivo.write("La cadena sobrescrita es: " + str(cad)
        + "\\n")
    archivo.write("ID"+ str(cont) + ".- (*,L,"+estado+")\\n
        \\n")

    cont = cont+1
elif(cad[i] == "|"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+"\\n")
    estado = "q8"
    cad[i] = "*"
    i = i-1
    print("El caracter sobre escrito es: *\n")
    archivo.write("El caracter sobre escrito es: *\n")
    print("(*,L,"+estado+")\\n")
    print(cad)
    archivo.write("La cadena sobrescrita es: " + str(cad)
        + "\\n")
    archivo.write("ID"+ str(cont) + ".- (*,L,"+estado+")\\n
        \\n")

    cont = cont+1
elif(cad[i] == "X"):
    print("Caracter analizado: " + cad[i] + "\n")
    archivo.write("Caracter analizado: "+cad[i]+"\\n")
    cad[i] = "*"
    print("\\nCadena Aceptada\\n")
    archivo.write("\\nCadena Aceptada\\n")
    print("(*,Z,F)+"\\n")
    print(cad)
    archivo.write("La cadena final es: " + str(cad) + "\\n
        ")
    archivo.write("ID"+ str(cont) + ".- (*,Z,F)+"\\n\\n")
    cont = cont+1
    break
else:
    estado = "q8"
    cad[i] = i
    print("(" + cad[i] + ",Z,"+estado+")\\n")
    print("Cadena Rechazada")
    break
except:
    print("Cadena Rechazada")

opc = 0
salir = 3
while opc != salir:
    print("\\n\\n*****Menu Principal*****\\n\\n")
    print("    ****Digite la opci n****")

```

```
opc = int(input(''))
1.- Cadena Manual
2.- Cadena Automática
3.- Salir
''')
if(opc == 1) or (opc == 2):
    maqTuring(opc)
elif opc == 3:
    print("Saliendo del Programa. Hasta Luego!!!!")
else:
    print("Opcion inválida, Vuelva a intentar")
```

Conclusiones

Para concluir con este noveno y último programa quiero destacar que intenté aplicar todos los conocimientos adquiridos a lo largo del curso, más que nada lo de los AFD.

Este programa en particular me resultó algo complicado de entender, ya que no lograba comprender al 100 por ciento el funcionamiento de una Máquina de Turing, sin embargo, después de revisar algunos tutoriales y repasar el PDF proporcionado por el profesor así como que de otros documentos en la red, logré comprender y realizar el análisis del programa.

En cuanto al programa en sí puedo concluir que este tema de la Máquina de Turing es de los más sino es que el más importante en la computación. Es fundamental la comprensión de este tema ya que a un nivel de extracción, llega a ser las bases de prácticamente todo.

Concluyo que ha sido un buen curso, a mi parecer al que más le he sacado provecho en cuanto a aprendizaje y en cuanto a práctica. No cabe duda que aun faltan muchas cosas por mejorar aprender y sobre todo de realizar un análisis previo para entender de lo que se está hablando.

Referencias

1. INAOE. (2017). Máquinas de Turing. Diciembre 21,2021, de INAOE Sitio web: <https://ccc.inaoep.mx/emorales/Cursos/Automatas/IntroMaquinas-Turing.pdf>
2. Bootcamp AI. (2021). Máquinas de Turing. Diciembre 21,2021, de Bootcamp AI Sitio web: <https://bootcampai.medium.com/>
3. UANL. (1996). MÁQUINAS DE TURING. Diciembre 21,2021, de UANL Sitio web: <http://www.itnuevolaredo.edu.mx/takeyas/apuntes/Automatas/Apuntes/turing.pdf>