

Universidad Rafael Landívar
Facultad de **Ingeniería**
Ingeniería en informática y sistemas
Curso: Lenguajes formales y autómatas
Sección: 01
Docente: Julio David Requena Duarte

Simulador máquina de Turing

Estudiante: Nery Samuel Hernández Herrera

Carné: 1098824

Guatemala, 08 de noviembre del 2025

ÍNDICE

DESCRIPCIÓN DEL PROGRAMA	1
II. INSTRUCCIONES DE INSTALACIÓN Y EJECUCIÓN.....	2
VI. EJECUCIÓN DEL PROGRAMA	3
II. DESCRIPCIÓN DE EXPRESIONES REGULARES	5

DESCRIPCIÓN DEL PROGRAMA

Qué es una Máquina de Turing

Una Máquina de Turing es un modelo matemático abstracto que representa un dispositivo de computación. Fue propuesto por Alan Turing en 1936 y es fundamental en la teoría de la computación.

Componentes de una Máquina de Turing

Cinta infinita: dividida en celdas, cada una contiene un símbolo.

Cabezal: lee y escribe símbolos en la cinta y se mueve hacia izquierda o derecha.

Estados: representan la configuración interna de la máquina.

Tabla de transiciones: define cómo la máquina cambia de estado según el símbolo leído.

Qué hace este simulador

El simulador de Máquina de Turing es una aplicación interactiva desarrollada en Python que permite visualizar el funcionamiento de una máquina de Turing en tiempo real, ejecutar máquinas paso a paso, validar cadenas de entrada según expresiones regulares y aprender conceptos de teoría de autómatas de forma interactiva.

Características principales

Diez expresiones regulares predefinidas, cada una con su propia máquina configurada.

Interfaz gráfica intuitiva con visualización clara de la cinta, cabezal y estados.

Modo paso a paso para aprendizaje detallado.

Modo automático con velocidad ajustable.

Historial completo de transiciones.

Código de colores: verde (aceptación), rojo (rechazo) y azul (procesando).

Componentes del programa

Clase TuringMachine: implementa la lógica de la máquina.

Transiciones: define las diez máquinas con sus reglas.

Interfaz gráfica: proporciona la parte visual interactiva.

Utilidades: funciones auxiliares para validación y pruebas.

II. INSTRUCCIONES DE INSTALACIÓN Y EJECUCIÓN

Requisitos del sistema:

Sistema operativo: Windows 10 o superior, macOS 10.14 o superior, o Linux Ubuntu 18.04 o superior.

Python: versión 3.8 o superior.

Espacio en disco: al menos 50 MB.

Memoria RAM mínima: 512 MB.

Verificación instalación de python

En Windows:

```
python --version
```

En macOS o Linux:

```
python3 --versión
```

Instalación paso a paso:

Paso 1: descargar o clonar el proyecto.

```
git clone [url-del-repositorio]
```

```
cd TercerProyecto
```

Paso 2: verificar estructura del proyecto:

```
TercerProyecto/
```

```
src/
```

```
turing_machine.py
```

```
transitions.py
```

```
gui.py
```

```
utils.py
```

```
tests/
```

```
docs/
```

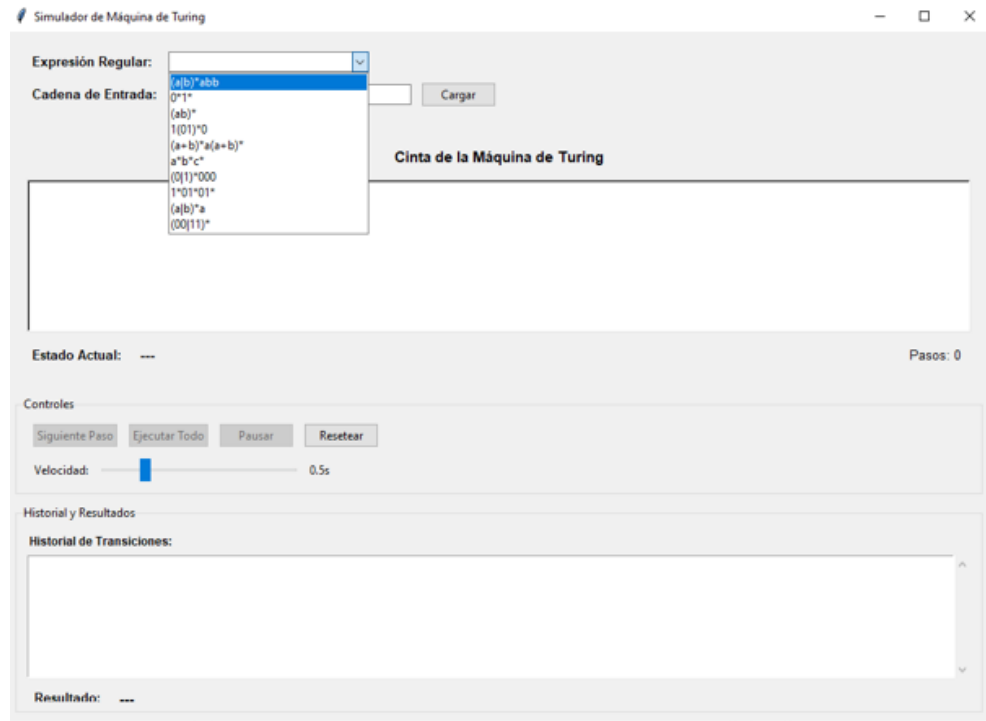
```
main.py
```

```
README.md
```

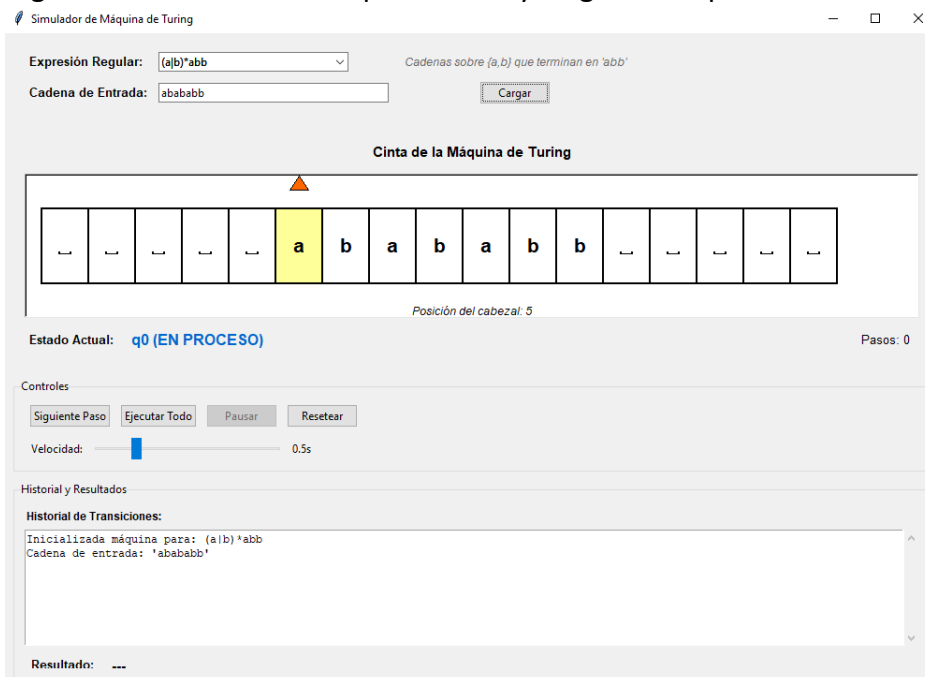
VI. EJECUCIÓN DEL PROGRAMA

Ejecutar el programa con su interfaz grafica mediante
python main.py o python src/gui.py

Selecciona una expresión regular del menú.



Ingresa la cadena en el campo de texto y cargar la maquina



Ejecuta paso a paso o de forma automática.

Simulador de Máquina de Turing

Expresión Regular: Cadenas sobre {a,b} que terminan en 'abb'

Cadena de Entrada:

Cinta de la Máquina de Turing

␣	a	b	a	b	a	b	b	␣	␣	␣	␣
---	---	---	---	---	---	---	---	---	---	---	---

Posición del cabezal: 12

Estado Actual: **qa (ACEPTACIÓN)** Pasos: 8

Controles

Velocidad: 0.5s

Historial y Resultados

Historial de Transiciones:

```
Inicializada máquina para: (a|b)*abb
Cadena de entrada: 'abababb'
Paso 1: Estado: q0 → q1, Leyó: 'a', Escribió: 'a', Movimiento: R
Paso 2: Estado: q1 → q2, Leyó: 'b', Escribió: 'b', Movimiento: R
Paso 3: Estado: q2 → q1, Leyó: 'a', Escribió: 'a', Movimiento: R
Paso 4: Estado: q1 → q2, Leyó: 'b', Escribió: 'b', Movimiento: R
Paso 5: Estado: q2 → q1, Leyó: 'a', Escribió: 'a', Movimiento: R
Paso 6: Estado: q1 → q2, Leyó: 'b', Escribió: 'b', Movimiento: R
Paso 7: Estado: q2 → q3, Leyó: 'b', Escribió: 'b', Movimiento: R
Paso 8: CADENA ACEPTADA
=== CADENA ACEPTADA ===
```

Resultado: **✓ CADENA ACEPTADA**

Al finalizar, se muestra si la cadena fue aceptada o rechazada.

II. DESCRIPCIÓN DE EXPRESIONES REGULARES

$(a|b)^*abb$

Acepta cualquier cadena sobre $\{a,b\}$ que termine con abb .

Ejemplos aceptados: abb , $aabb$, $bababb$, $abbabb$.

Ejemplos rechazados: ab , aba , ba , cadena vacía.

01

Acepta cadenas con cualquier cantidad de 0s seguidos de 1s.

Ejemplos aceptados: 0 , 1 , 001 , 000111 .

Ejemplos rechazados: 10 , 010 , 01 .

$(ab)^*$

Acepta repeticiones del patrón ab .

Ejemplos aceptados: ab , $abab$, $ababab$.

Ejemplos rechazados: a , b , aba , ba .

$1(01)^*0$

Empieza con 1 y termina con 0.

Ejemplos aceptados: 10 , 1010 , 101010 .

Ejemplos rechazados: 1 , 0 , 101 .

$(a+b)a(a+b)$

Contiene al menos una a .

Ejemplos aceptados: a , ba , ab , bab .

Ejemplos rechazados: b , bb , bbb , vacía.

abc^*

Bloques de letras en orden a , b , c .

Ejemplos aceptados: a , ab , abc , $aabbcc$.

Ejemplos rechazados: ac , ba , cb .

$(0|1)^*000$

Termina con 000.

Ejemplos aceptados: 000, 1000, 01000.

Ejemplos rechazados: 00, 001, 100.

10101^*

Contiene exactamente dos ceros.

Ejemplos aceptados: 00, 010, 001, 1001.

Ejemplos rechazados: 0, 000, 111.

$(a|b)^*a$

Termina en a.

Ejemplos aceptados: a, ba, aa, abababa.

Ejemplos rechazados: b, bb, ab.

$(00|11)^*$

Cadenas formadas por pares iguales.

Ejemplos aceptados: 00, 11, 0011, 1100.

Ejemplos rechazados: 0, 1, 01, 10, 001.