

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 2

CATEDRÁTICO: ING. DAVID ESTUARDO MORALES AJCOT

TUTOR ACADÉMICO: HERBERTH ABISAI AVILA RUIZ



## **Manual de usuario**

Josué Daniel Fuentes Díaz

CARNÉ: 202300668

SECCIÓN: B+

GUATEMALA, 28 DE MARZO DEL 2,025

## ÍNDICE

<b>ÍNDICE</b>	<b>1</b>
<b>OBJETIVOS DEL SISTEMA</b>	<b>2</b>
GENERAL	2
ESPECÍFICOS	2
<b>INTRODUCCIÓN</b>	<b>3</b>
<b>FLUJO DE LAS FUNCIONALIDADES DEL SISTEMA</b>	<b>4</b>

## **OBJETIVOS DEL SISTEMA**

### **GENERAL**

Guiar al usuario en el uso correcto del sistema AFDGraph, explicando paso a paso cómo analizar archivos .lfp, visualizar autómatas y generar reportes léxicos.

### **ESPECÍFICOS**

- Explicar la interfaz y los componentes principales del programa para facilitar su comprensión y manejo.
- Instruir al usuario sobre cómo cargar archivos, seleccionar AFDs y generar reportes de tokens y errores léxicos de manera automática.

# INTRODUCCIÓN

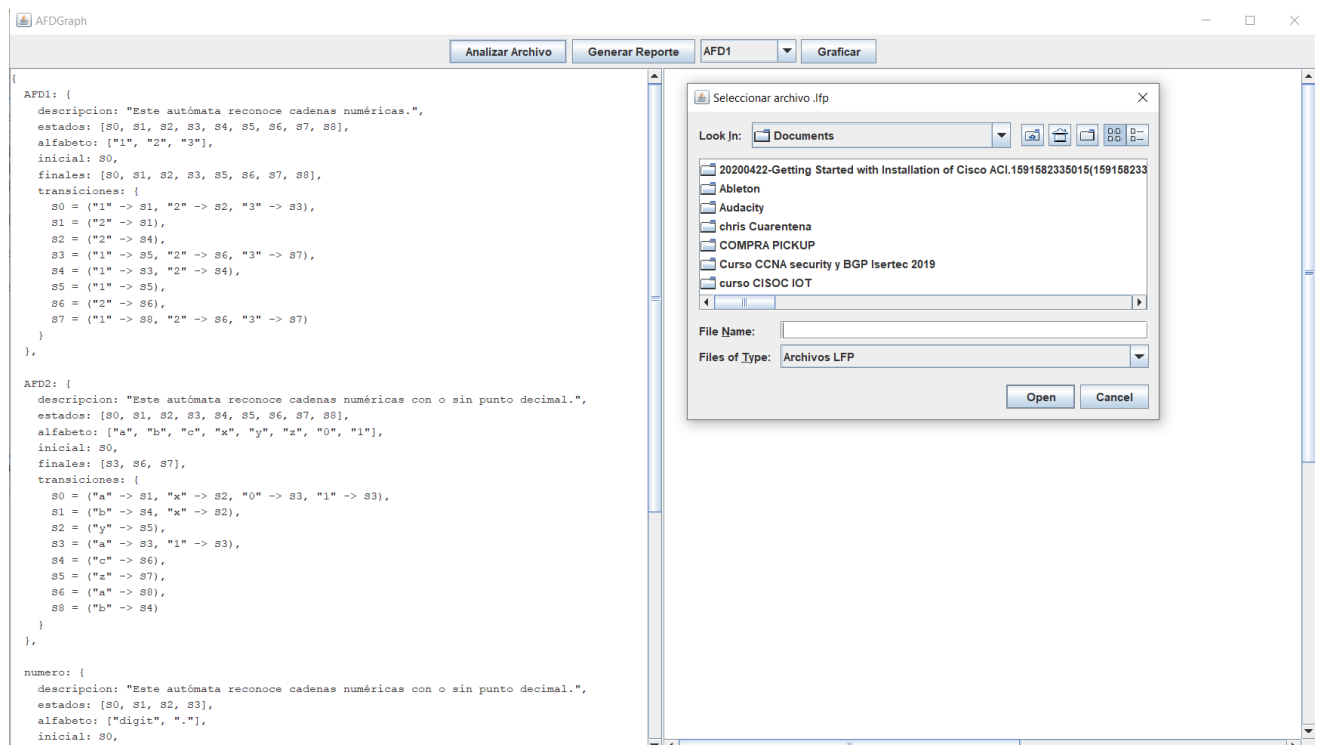
AFDGraph es una aplicación de escritorio desarrollada en Java que permite al usuario analizar archivos con definiciones de autómatas finitos deterministas (.lfp), visualizar gráficamente los AFDs definidos y generar reportes detallados de tokens y errores léxicos. La interfaz gráfica amigable facilita la carga, análisis y exploración de múltiples AFDs, ayudando tanto en la validación como en la interpretación de sus estructuras.

Este manual proporciona una guía clara para utilizar correctamente todas las funcionalidades del sistema.

# FLUJO DE LAS FUNCIONALIDADES DEL SISTEMA

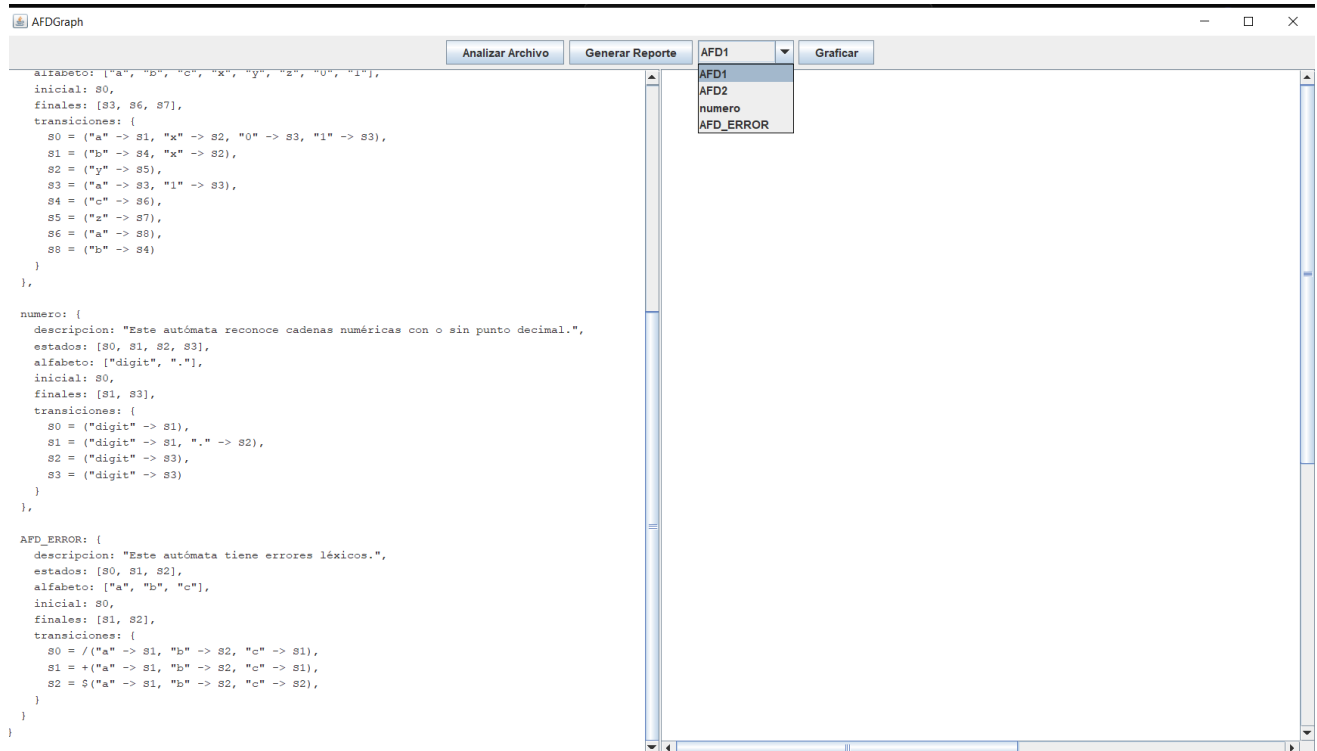
El uso del programa AFDGraph sigue un flujo sencillo e intuitivo:

- **Carga del archivo .lfp:**
  - El usuario selecciona el botón "**Analizar Archivo**".
  - Se abre un explorador de archivos para elegir un archivo con extensión .lfp.
  - El sistema analiza el contenido, valida la estructura de cada AFD y muestra un resumen.



- **Visualización del contenido:**

- El contenido original del archivo se muestra en un área de texto de solo lectura.
- Se actualiza una lista desplegable con los nombres de los AFDs detectados



- **Selección y graficación del AFD:**

- El usuario selecciona un AFD desde el combo box
- Al presionar el botón "Graficar", el sistema valida el AFD seleccionado
  - Si es válido, se genera y muestra su representación gráfica
  - Si tiene errores, se muestra una ventana detallada con los errores encontrados

Analizar Archivo   Generar Reporte   AFD1   Graficar

```
AFD1: {
  descripcion: "Este autómata reconoce cadenas numéricas.",
  estados: [S0, S1, S2, S3, S4, S5, S6, S7, S8],
  alfabeto: ["1", "2", "3"],
  inicial: S0,
  finales: [S0, S1, S2, S3, S5, S6, S7, S8],
  transiciones: {
    S0 = ("1" -> S1, "2" -> S2, "3" -> S3),
    S1 = ("2" -> S1),
    S2 = ("2" -> S4),
    S3 = ("1" -> S5, "2" -> S6, "3" -> S7),
    S4 = ("1" -> S3, "2" -> S4),
    S5 = ("1" -> S5),
    S6 = ("2" -> S6),
    S7 = ("1" -> S8, "2" -> S6, "3" -> S7)
  }
},
AFD2: {
  descripcion: "Este autómata reconoce cadenas numéricas con o sin punto decimal.",
}
```

Diagrama de AFD1:

```
graph LR
  S0((S0)) -- 1 --> S1((S1))
  S0 -- 2 --> S2((S2))
  S0 -- 3 --> S3((S3))
  S1 -- 2 --> S1
  S2 -- 2 --> S4((S4))
  S3 -- 1 --> S5((S5))
  S3 -- 2 --> S6((S6))
  S3 -- 3 --> S7((S7))
  S4 -- 1 --> S3
  S4 -- 2 --> S4
  S5 -- 1 --> S5
  S6 -- 2 --> S6
  S7 -- 1 --> S8((S8))
  S7 -- 2 --> S6
  S7 -- 3 --> S7
```

**Errores en AFD AFD\_ERROR**

El AFD 'AFD\_ERROR' tiene 4 errores:

1. Transición sin paréntesis válidos: S0 = /("a" -> S1, "b" -> S2, "c" -> S1)
2. Transición sin paréntesis válidos: S1 = +("a" -> S1, "b" -> S2, "c" -> S1)
3. Transición sin paréntesis válidos: S2 = \$("a" -> S1, "b" -> S2, "c" -> S2)
4. No hay transiciones definidas para el estado 'S0'

No se puede graficar hasta corregir los errores.

OK

- **Generación de reportes:**

- Al hacer clic en "Generar Reporte", se crean automáticamente dos archivos HTML
  - **Reporte de Tokens:** Lista todos los tokens válidos encontrados durante el análisis
  - **Reporte de Errores Léxicos:** Muestra los errores léxicos detectados, indicando línea, columna y tipo
- Los archivos se guardan en la carpeta reportes/ dentro del directorio del proyecto

#### Reporte de Tokens

Tipo	Lexema	Línea	Columna
SIMBOLO	{	1	1
IDENTIFICADOR	AFD1	1	9
PALABRA_RESERVADA	descripcion	1	28
IDENTIFICADOR	ómata	1	44
IDENTIFICADOR	noce	1	53
IDENTIFICADOR	enas	1	61
IDENTIFICADOR	éricas	1	71
PALABRA_RESERVADA	estados	1	86
IDENTIFICADOR	S1	1	95
IDENTIFICADOR	S2	1	99
IDENTIFICADOR	S3	1	103
IDENTIFICADOR	S4	1	107
IDENTIFICADOR	S5	1	111
IDENTIFICADOR	S6	1	115
IDENTIFICADOR	S7	1	119
IDENTIFICADOR	S8	1	123
PALABRA_RESERVADA	alfabeto	2	13
CADENA	"2"	2	21

#### Reporte de Errores Léxicos

Descripción	Lexema	Línea	Columna
Símbolo '/' no válido	(	22	33
Carácter no válido: '+'	+	22	79
Carácter no válido: '\$'	\$	22	126