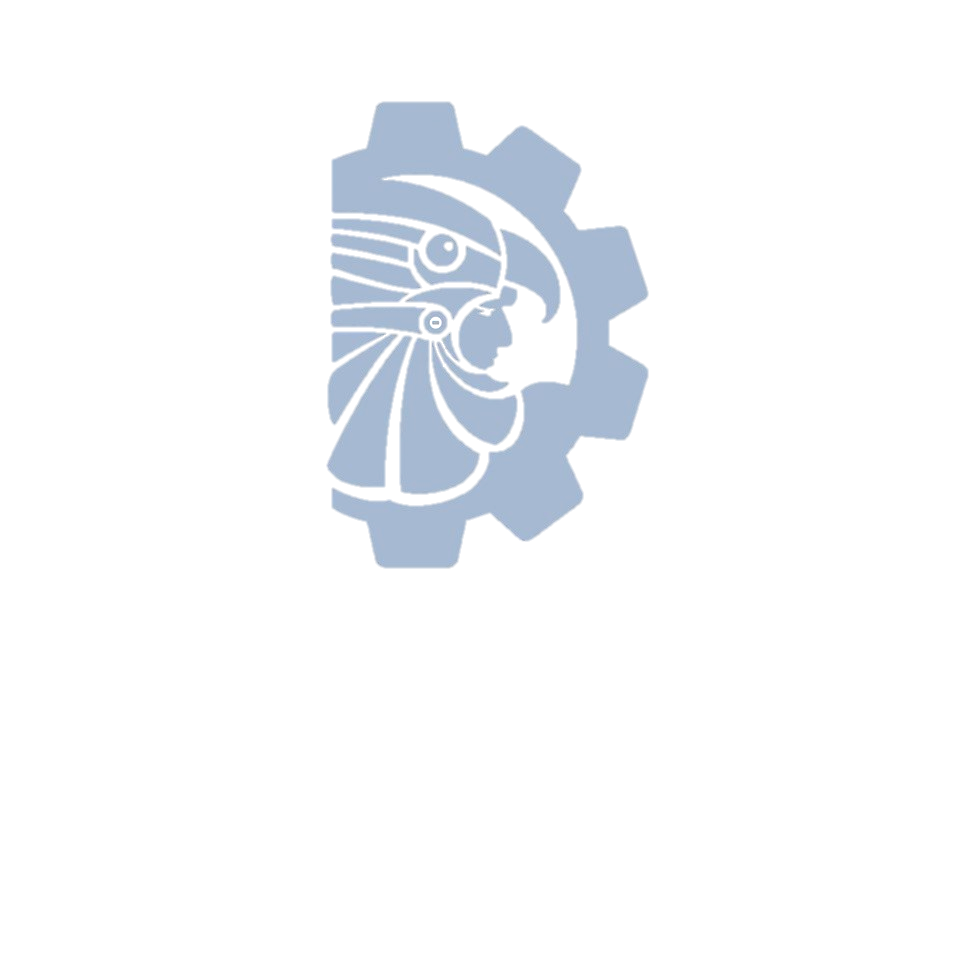




**Documentación ER y AF para Analizador Léxico**

**INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**NOMBRE DE LA ASIGNATURA**

Lenguajes y autómatas I

**Docente:**

Rodolfo Baume Lazcano

**Alumnos:**

Vera Rodriguez Hector Gabriel

García Lugo Itzel Paola

Diego Quintos Cabeza

**Fecha:** 20/05/2024

**Definición del lenguaje:**

* Palabras clave ( if, else, while)
* Identificadores (nombres de variables y funciones)
* Operadores (+,-, \*,/)
* Números (constantes numéricas)
* Símbolos ( (, ), {, })
* Espacios en blanco y comentarios (que generalmente se ignoran)

**Especificación de tokens:**

* Palabras clave:

if

else

while

* Identificadores:

[a-zA-Z\_][a-zA-Z0-9\_]\*

* Operadores:

\+, -, \\*, /

* Números:

\d+

* Símbolos:

\(, \), {, },

* Espacios en blanco y comentarios: (que generalmente se ignoran)

**Expresiones regulares:**

* Identificadores: [a-zA-Z\_][a-zA-Z0-9\_]\*
* Números: \d+
* Operadores: \+, -, \\*, /
* Símbolos: \(, \), {, }

**Tabla de tokens**

| Token | Clasificación |
| --- | --- |
| if | Palabra clave |
| else | Palabra clave |
| while | Palabra clave |
| + | Operador |
| - | Operador |
| \* | Operador |
| / | Símbolos |
| ( | Símbolos |
| ) | Símbolos |
| { | Símbolos |
| } | Símbolos |
| 0-9 | Números |
|  | Espacio en blanco |

**1.- Palabras clave:**

Las palabras clave nos indican tokens que serán reservados para realizar operaciones específicas

* If: un token condicional que si algo sucede entonces el programa realizará una acción
* Else: este token lo utilizaremos en conjunto con el token if el cual nos indica una condición si la condición no se cumple entra en juego este token que si la acción declarada en el if no sucede entonces sucederá otra acción.
* While: El token While nos indica que se generará un ciclo que mientras se cumpla una condición se repetirá una acción hasta que la condición ya no cumpla con los requerimientos

| Token | Clasificación | Expresión regular |
| --- | --- | --- |
| If | Palabra clave | \bif \b |
| Else | Palabra clave | \belse \b |
| While | Palabra clave | \bwhile \b |

**2.- Identificador**

Estos serán utilizados para nombrar las variables, funciones, clases, etc, elementos que serán definidos por el usuario serán palabras que estarán formadas por letras mayúsculas, minúsculas y guión bajo junto con números.

| Token | Clasificación | Expresión regular |
| --- | --- | --- |
| Identificador | Identificadores | [a-zA-Z\_][a-zA-Z0-9\_]\* |

**3.- Operador**

Los tokens de tipo operado serán operadores aritméticos que nos permitirán realizar las principales operaciones aritméticas:

* Suma: se realizará la suma de dos números
* Resta:se realizará la resta de un número mayor y uno menor
* Multiplicación: Se realiza la multiplicación de 2 números.
* División:se realizará la división de 2 dígitos.

| Token | Clasificación | Expresión regular |
| --- | --- | --- |
| Operador | Operadores | \+, -, \\*, / |

**4.- Símbolo:**

Los símbolos utilizados en este lenguaje nos servirán para identificar o definir uno o varios token dentro de una línea de código para encerrarlos y que estos se conviertan en un solo token.

| Token | Clasificación | Expresión regular |
| --- | --- | --- |
| Símbolo | Símbolos | \(, \), {, } |

**5.- Número:**

Los números serán nuestros principales tokens para realizar las acciones en conjunto de nuestros operadores y nuestras palabras reservadas, podemos encerrar números dentro de los símbolos y realizar operaciones.

* 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

| Token | Clasificación | Expresión regular |
| --- | --- | --- |
| Símbolo | Símbolos | \d+ |

**6.- Espacios en blanco**

Los espacios en blanco serán ignorados y no se marcan dentro del analizador ya que seran espacio sin información

| Token | Clasificación | Expresión regular |
| --- | --- | --- |
| Espacio | Espacios en blanco |  |