

# ANALIZADOR LÉXICO TINY(0)

Grupo 29

Adrian Estevez Gallego

Alvaro Delgado Gutierrez

## 1. Identificación y definición informal de las clases léxicas

Suponemos que el lenguaje diferencia entre mayúsculas y minúsculas.

### Clases Léxicas

- **EntNum**: Clase léxica que identifica los números enteros. Los números enteros empiezan opcionalmente por un signo + o -. Seguidamente debe aparecer una secuencia de uno o más dígitos, no se admiten 0s no significativos a la izquierda.
- **RealNum**: Clase léxica que identifica los números reales. Los números reales son aquellos con una parte entera (misma especificación que EntNum) y adicionalmente cuentan con una parte decimal y/o una parte exponencial.
- **Variable**: Clase léxica que identifica los nombres de variables. Estos deben comenzar por una letra seguida de una secuencia de 0 o más letras, dígitos o subrayados (“\_”).
- **True** : Clase léxica que identifica la palabra reservada para un booleano cierto.
- **False** : Clase léxica que identifica la palabra reservada para un booleano falso.
- **Bool**: Clase léxica que identifica la palabra reservada “*bool*” para el tipo booleano.
- **Int**: Clase léxica que identifica la palabra reservada “*int*” para el tipo entero.
- **Real** : Clase léxica que identifica la palabra reservada “*real*” para el tipo de números reales.
- **Sum** : Clase léxica que identifica la suma de dos operandos. Ambos operandos deben ser números (EntNum o RealNum), uno de ellos a la izquierda y otro a la derecha de este.
- **Res** : Clase léxica que identifica la resta entre dos operandos. Ambos operandos deben ser números (EntNum o RealNum). El minuendo a la izquierda de este y el sustraendo a la derecha.
- **Mul** : Clase léxica que identifica la multiplicación de dos operandos. Ambos operandos deben ser números (EntNum o RealNum), uno de ellos a la izquierda y otro a la derecha de este.

- **Div** : Clase léxica que identifica la división entre dos operandos. Ambos operandos deben ser números (EntNum o RealNum). El dividendo a la izquierda del este y el divisor a la derecha.
- **Asig** : Clase léxica que identifica la instrucción de asignación a una variable designada por el símbolo “=”.
- **Menor** : Clase léxica que identifica el operador relacional “<”.
- **Mayor** : Clase léxica que identifica el operador relacional “>”.
- **Ig** : Clase léxica que identifica el operador relacional “==”.
- **MayIg** : Clase léxica que identifica el operador relacional “>=”.
- **MenIg** : Clase léxica que identifica el operador relacional “<=”.
- **Dist** : Clase léxica que identifica el operador relacional “!=”.
- **And** : Clase léxica que identifica el operador lógico “and”.
- **Or** : Clase léxica que identifica el operador lógico “or”.
- **Not** : Clase léxica que identifica el operador lógico “not”.
- **PAp** : Clase léxica que identifica la apertura de un paréntesis con el símbolo “(”.
- **PCie** : Clase léxica que identifica el cierre de un paréntesis con el símbolo “)”.
- **PunCo** : Clase léxica que identifica el final de una sentencia designado por “;”.
- **Sep** : Clase léxica que identifica el separador entre la sección de declaraciones y la sección de instrucciones. Este separador es “&&”.

### Cadenas ignorables

- Espacio
- Tabulación
- Backspace
- Salto de línea
- Retorno de carro

## 2. Especificación formal de las clases léxicas

### Definiciones auxiliares

- DígitoPos = [1-9]
- Dígito = (0 | [1-9])
- Ent = ((DígitoPos Dígito\*) | 0)
- PDec = . (Dígito\* DígitoPos | 0)
- PExp = (e | E) (\- | \+)? ( (DígitoPos Dígito\*) | 0 )
- letra = [a-z, A-Z]

### Clases léxicas

- EntNum = ( \- | \+ )? ( Ent | 0 )
- RealNum = ( \- | \+ )? ( Ent | 0 ) ( PDec | PExp | PDec PExp )
- Variable = letra ( letra | Dígito | \\_ )\*
- True = true
- False = false
- Bool = bool
- Int = int
- Real = real
- Sum = \+
- Res = \-
- Mul = \\*
- Div = \
- Asig = \=
- Menor = <
- Mayor = >
- Ig = \= \=
- Maylg = > \=
- Menlg = < \=
- Dist = ! \=
- And = and
- Or = or

- Not = not
- PAp = \(\)
- PCie = \(\)
- PunCo = ;
- Sep = &&

## Cadenas ignorables

- Esp = <espacio>
- Salto = \n
- Tab = \t
- Back = \b
- RetCar = \r

### 3. Diagrama de transiciones

