

## PROGRAMA N°1

Profesora : Consuelo Ramírez  
Fecha : 1 de septiembre  
Puntaje : 61 puntos  
PREMA : 60%

### Objetivo

Implementar un analizador léxico, utilizando un lenguaje de programación.

### Requerimientos

Escriba un programa en lenguaje de programación C para sistema operativo Windows que implemente un analizador léxico para el lenguaje de programación descrito.

Su programa debe:

1. permitir ingresar un archivo de entrada (programa fuente) y un archivo de salida como parámetros en la línea de comandos.
2. escribir en mayúsculas en el archivo de salida cada uno de los respectivos componentes léxicos leídos del programa fuente seguidos de un salto de línea.

La Figura 1 representa la forma de ejecución del programa.

```
C:\> lexico.exe archivo_entrada archivo_salida  
C:\>
```

**Figura 1.** Ejecución.

Las Figuras 2, 3, 4, 5 y 6 representan los tipos de errores que el programa debe detectar.

```
C:\>lexico.exe  
Error: Faltan parámetros.  
Uso: lexico.exe archivo_entrada archivo_salida  
C:\>
```

**Figura 2.** Error 1.

```
C:\> lexico.exe archivo
Error: Falta parámetro.
Uso: lexico.exe archivo_entrada archivo_salida

C:\>
```

**Figura 3.** Error 2.

```
C:\> lexico.exe archivo_1 archivo_2 archivo_3 ...
Error: Demasiados parámetros.
Uso: lexico.exe archivo_entrada archivo_salida

C:\>
```

**Figura 4.** Error 3.

```
C:\> lexico.exe archivo_entrada archivo_salida
Error: El archivo de entrada no existe.

C:\>
```

**Figura 5.** Error 4.

```
C:\> lexico.exe archivo_entrada archivo_salida
Error: El archivo de salida ya existe.

C:\>
```

**Figura 6.** Error 5.

Cuando ocurre algún error no se debe generar un archivo de salida.

## Entrega

1. Comprima el archivo fuente en un archivo con formato Zip denominado con su primer nombre seguido de un espacio y de su primer apellido.
2. Suba el archivo resultante a Campus Virtual antes del 29 de septiembre a las 11:10 hrs.

## Lenguaje de programación

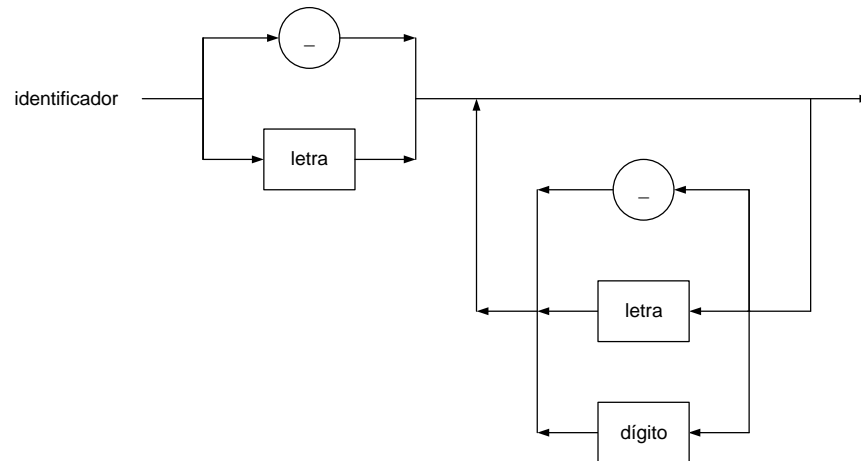
Palabras reservadas:

- |             |            |          |
|-------------|------------|----------|
| • program   | • function | • for    |
| • const     | • begin    | • to     |
| • type      | • end      | • downto |
| • var       | • if       | • repeat |
| • integer   | • then     | • until  |
| • real      | • else     | • or     |
| • boolean   | • case     | • div    |
| • char      | • of       | • mod    |
| • procedure | • while    | • and    |
| • forward   | • do       | • not    |

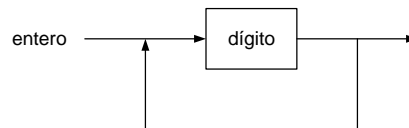
Símbolos:

- ;
- .
- =
- :
- ,
- +
- -
- '
- (
- )
- <
- >
- \*
- /
- .. (rango)
- := (asignación)
- <= (menor\_igual)
- <> (distinto)
- >= (mayor\_igual)

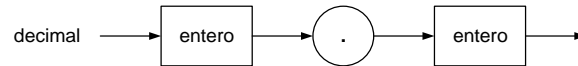
Las Figuras 7, 8, 9 y 10 muestran la estructura de los identificadores, enteros, reales en notación decimal y reales en notación exponencial respectivamente.



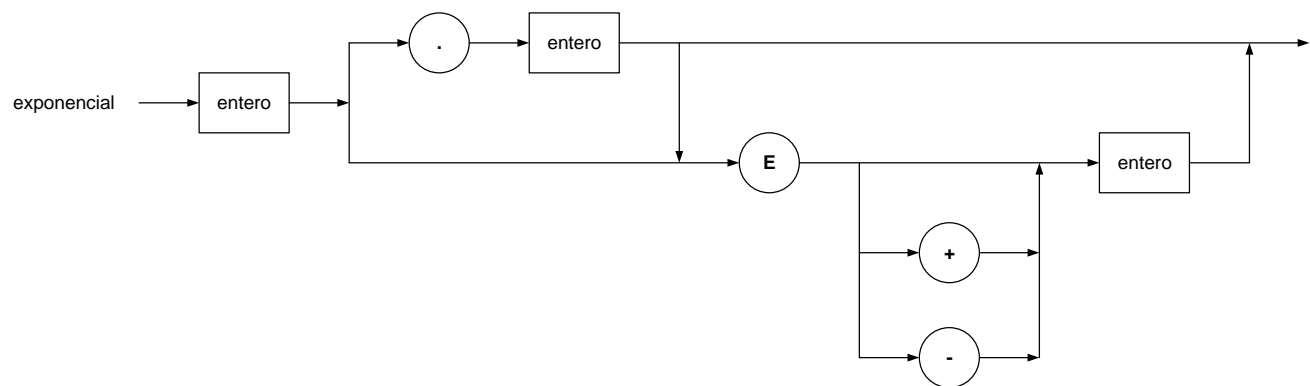
**Figura 7.** Identificador.



**Figura 8.** Entero.



**Figura 9.** Real en notación decimal.



**Figura 10.** Real en notación exponencial.