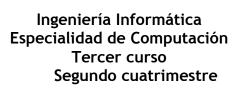


# PROCESADORES DE LENGUAJE





Departamento de Informática y Análisis Numérico Escuela Politécnica Superior de Córdoba Universidad de Córdoba

Curso académico 2016 - 2017

# Hoja de ejercicios de FLEX

#### 1. Sumar

- Codifica un programa en flex que copie el archivo de entrada en uno de salida, poniendo a 0 a todo número positivo que sea múltiplo de 3.
- Ejemplo

# Entrada.txt

1	2	3	4
5	6	7	8
9	10	11	12

### Salida.txt

1	2	0	4
5	0	7	8
Ω	10	11	Ω

- Observación:
  - Se recomienda usar la función atoi() de C que transforma una cadena de caracteres en su valor entero.

### 2. Sumar parametrizado

- Codifica un programa en flex que copie el archivo de entrada en uno de salida, sumando N1 a todo número positivo que sea múltiplo de N2, donde N1 y N2 son dos números pasados como argumentos desde la línea de órdenes.
- Ejemplo: ./sumar\_parametrizado.exe Entrada.txt Salida.txt 5 4

### Entrada.txt

5	6	./	8
9	10	11	12
Salid	da.txt		
1	2	3	9
5	6	7	13
9	10	11	17

### Observación:

 Se recomienda usar la función atoi() de C que transforma una cadena de caracteres en su valor entero.

### 3. Contar

• Elaborar un programa de flex que reciba un fichero de texto y cuente el número de caracteres, palabras y líneas que contiene.

# 4. Contar apariciones de una palabra

 Codifica un programa flex que reciba un fichero de texto y una palabra y cuente el número veces que aparece dicha palabra en el fichero.

# 5. Sustitución de una palabra

- Codifica un analizador que reemplace una palabra por otra en un fichero de entrada.
- Ambas palabras, así como el nombre del fichero deberán ser introducidos por el usuario, bien a través de la línea de comandos o cuando el usuario ejecute el programa.

# 6. Fichero de registro de acceos

- Considérese un fichero de registro que contiene información sobre los accesos de los usuarios a un sistema informático.
- Cada línea tiene la siguiente información
  - o usuario fecha hora
- Codifica un analizador léxico que permita
  - o Contar cuántos accesos ha realizado un usuario
    - El programa recibirá como argumentos
      - ✓ el nombre del fichero de registro
      - ✓ el nombre del usuario
  - Contar cuántos trabajos accesos ha realizado un usuario en un día concreto
    - el nombre del fichero de registro
    - el nombre del usuario
    - el día que se desea consultar.

# 7. Analizador léxico de pseudocódigo

• Codifica un analizador léxico que permita reconocer los

componentes léxicos de un programa escrito en pseudocódigo.

#### Palabras reservadas

- o inicio, fin, \_\_mod, \_\_o, \_\_y, \_\_no, leer, escribir, si, entonces, si\_no, fin\_si, mientras, hacer, fin\_mientras, repetir, hasta\_que, para, desde, hasta, paso, fin\_para.
- No se distinguirá entre mayúsculas ni minúsculas.
- Las palabras reservadas no se podrán utilizar como identificadores.

#### Identificador

- Características
  - Estarán compuestos por una serie de letras, dígitos y el subrayado;
  - Deben comenzar por una letra,
  - No podrán acabar con el símbolo de subrayado, ni tener dos subrayados consecutivos.
  - No se distinguirá entre mayúsculas ni minúsculas.
- Ejemplos
  - Identificadores válidos:

dato, dato 1, dato 1 a

Identificadores no válidos:

\_dato, dato\_, dato\_\_1

### Número

- Se utilizarán números enteros, reales de punto fijo y reales con notación científica.
- o Todos ellos serán tratados conjuntamente como números.

### Cadena

 Estará compuesta por una serie de caracteres delimitados por comillas simples:

'Ejemplo de cadena'

 Deberá permitir la inclusión de la comilla simple utilizando la barra (\):

'Ejemplo de cadena con \' comillas\' simples'.

- Nota:
  - Las comillas exteriores no formarán parte de la cadena.

# • Operador de asignación

ASIGNACIÓN: :=

# • Operadores aritméticos:

- o SUMA: +
- o RESTA: -
- o PRODUCTO: \*

DIVISIÓN: /MÓDULO: \_\_modPOTENCIA: \*\*

• Operador alfanumérico:

CONCATENACIÓN: |

- Operadores relacionales de números y cadenas:
  - MENOR\_QUE: <</li>
    MENOR\_IGUAL\_QUE: <=</li>
    MAYOR\_QUE: >=
    MAYOR\_IGUAL\_QUE: >=
    IGUAL: ==
    DISTINTO: <>
  - o Por ejemplo:
    - Si A es una variable numérica y control una variable alfanumérica, se pueden generar las siguientes expresiones relacionales:

- Operadores lógicos:
  - DISYUNCIÓN\_LÓGICA: \_\_oCONJUNCIÓN LÓGICA: v
  - NEGACIÓN LÓGICA: no
  - Por ejemplo:

- Comentarios
  - De varias líneas: delimitados por (\* y \*)

```
(* ejemplo maravilloso
de comentario
de tres líneas *)
```

- De una línea:
  - Todo lo que siga al carácter # hasta el final de la línea.

# ejemplo espectacular de cometario de una línea

- Otro componentes léxicos
  - o FIN\_SENTENCIA: ;
  - Paréntesis
    - Izquierdo: (
    - Derecho: )

### • Control de errores

- o El intérprete deberá controlar toda clase de errores:
  - Identificador mal escrito.
  - Números mal escritos.
  - Utilización de símbolos no permitidos.
  - Etc.

#### Prueba

- Se deberá comprobar el funcionamiento del analizador léxico usando tres ficheros:
  - Fichero denominado Newton.txt
  - ejemplo\_1.txt: fichero original sin errores.
  - ejemplo\_2.txt: fichero original **con** errores.