



PROCESADORES DE LENGUAJE

Ingeniería Informática
Especialidad de Computación
Tercer curso
Segundo cuatrimestre



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

Curso académico 2018 - 2019

Hoja de ejercicios de FLEX

1. Eliminar múltiplos

- Codifica un programa en flex que
 - reciba un número entero N y el nombre de un fichero compuesto por números enteros
 - genere otro fichero con los números que no sean múltiplos del número N

- Ejemplo:

./eliminar_mutiplos 3 entrada.txt salida.txt

Entrada.txt

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

Salida.txt

1	2	4
5	7	8
10	11	

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

2. Sustitución de una palabra

- Codifica un analizador que
 - Reemplace una palabra por otra en un fichero de entrada.
 - Al final, se debe contar el número de palabras sustituidas

- Ejemplo:

./sustituir.exe fichero.txt antigua nueva

3. Analizador léxico de pseudocódigo

- Codifica un analizador léxico que permita reconocer los componentes léxicos de un programa escrito en pseudocódigo.
- **Palabras reservadas**
 - *inicio, fin, mod, 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.
 - 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:**
 - SUMA: +

- RESTA: -
- PRODUCTO: *
- DIVISIÓN: /
- MÓDULO: mod
- POTENCIA: **
- **Operador alfanumérico:**
 - CONCATENACIÓN: ||
- **Operadores relacionales de números y cadenas:**
 - MENOR_QUE: <
 - MENOR_IGUAL_QUE: <=
 - MAYOR_QUE: >
 - MAYOR_IGUAL_QUE: >=
 - IGUAL: ==
 - DISTINTO: <>
 - Por ejemplo:
 - Si A es una variable numérica y *control* una variable alfanumérica, se pueden generar las siguientes expresiones relacionales:
(A >= 0)
(control <> 'stop')
- **Operadores lógicos:**
 - DISYUNCIÓN_LÓGICA: .o.
 - CONJUNCIÓN_LÓGICA: .y.
 - NEGACIÓN_LÓGICA: .no.
 - Por ejemplo:
(A >= 0) .y. .no. (control <> 'stop')
- **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 comentario de una línea
- **Otro componentes léxicos**
 - FIN_SENTENCIA: ;
 - Paréntesis
 - Izquierdo: (

- Derecho:)

- **Control de errores**

- 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.

4. Registros de impresión de documentos

- Considérese un fichero que contiene los registros con la información de los impresos por un usuario en una impresora.
- Cada línea tiene la siguiente información
 - Usuario documento fecha hora
 - Observación
 - Cada estudiante debe elegir el formato de los usuarios, la fecha y la hora.
- Codifica un analizador léxico que permita
 - Contar cuántos documentos ha impreso un usuario
 - El programa recibirá como argumentos
 - ✓ el nombre del fichero de registro
 - ✓ el nombre del usuario
 - Ejemplo
contar fichero usuario
 - Contar cuántos trabajos ha impreso un usuario en un día concreto
 - el nombre del fichero de registro
 - el nombre del usuario
 - el día que se desea consultar.
 - Ejemplo
contar fichero usuario fecha
 - Cualquier otra opción que se considere interesante
- Observación
 - Se recomienda el uso de “Estados de Flex”.

5. Cifrado de Vigenère

- Codificar un analizador léxico que permite cifrar y descifrar un texto usando el cifrado de Vigenère
- Este método es una generalización del cifrado César
- Observaciones
 - Se valorará que se permita usar
 - letras mayúsculas y minúsculas.
 - la letra “ñ”
 - letras acentuadas.
 - Los caracteres que no sean alfabéticos no se modificarán
- Referencias del cifrado Vigenère
 - Universidad Politécnica de Madrid
 - http://www.dma.fi.upm.es/recursos/aplicaciones/matematica_discreta/web/aritmetica_modular/polialfabeto.html
 - Universidad de Granada
 - <https://www.ugr.es/~anillos/textos/pdf/2011/EXPO-1.Criptografia/02a11.htm>
 - La cifra Vigenère: el misterioso código que se tardó tres siglos en descifrar. Diario ABC. 20 de noviembre de 2018
 - https://www.abc.es/ciencia/abci-cifra-vigenere-misterioso-codigo-tardo-tres-siglos-descifrar-201811192150_noticia.html
 - Criptografía (XIX): cifrado Vigenère y criptoanálisis Kasiski (II). Blog de Mikel García Arragán.
 - <http://mikelgarcialarragan.blogspot.com/2015/11/criptografia-xix-cifrado-vigenere-y.html>
 - Wikipedia
 - https://es.wikipedia.org/wiki/Cifrado_de_Vigen%C3%A8re
 - Antecedente: Cifrado César
 - Matemáticas cercanas. Rompiendo los muros de las matemáticas
 - ✓ <https://matematicascercanas.com/2017/11/26/cifrado-sustitucion-codigo-cesar/>
 - Wikipedia
 - ✓ https://es.wikipedia.org/wiki/Cifrado_C%C3%A9sar