

PRÁCTICA 2

Recuperación de la información



27 DE OCTUBRE DE 2018
VÍCTOR ANDRÉS VIZÁN

Contenido

1.	Introducción	2
2.	Desarrollo	3
3.	Pruebas.....	7
4.	Conclusiones y observaciones.....	9
5.	Bibliografía	10

1. Introducción

Esta segunda práctica de la asignatura usa los códigos de la anterior práctica para hacer una consulta al programa mediante el método booleano.

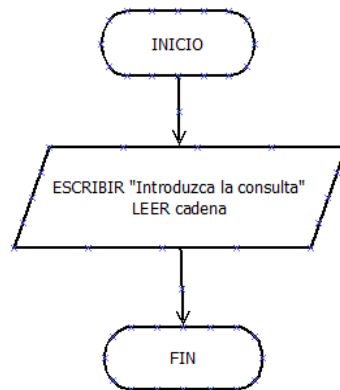
El programa se estructura en diversos puntos:

- Seguir un menú con diferentes opciones.
- La primera opción del menú te permite normalizar documento como en la práctica 1.
- La segunda opción te permite hacer una consulta mediante el método booleano introduciendo la consulta y el nombre de los documentos por teclado.

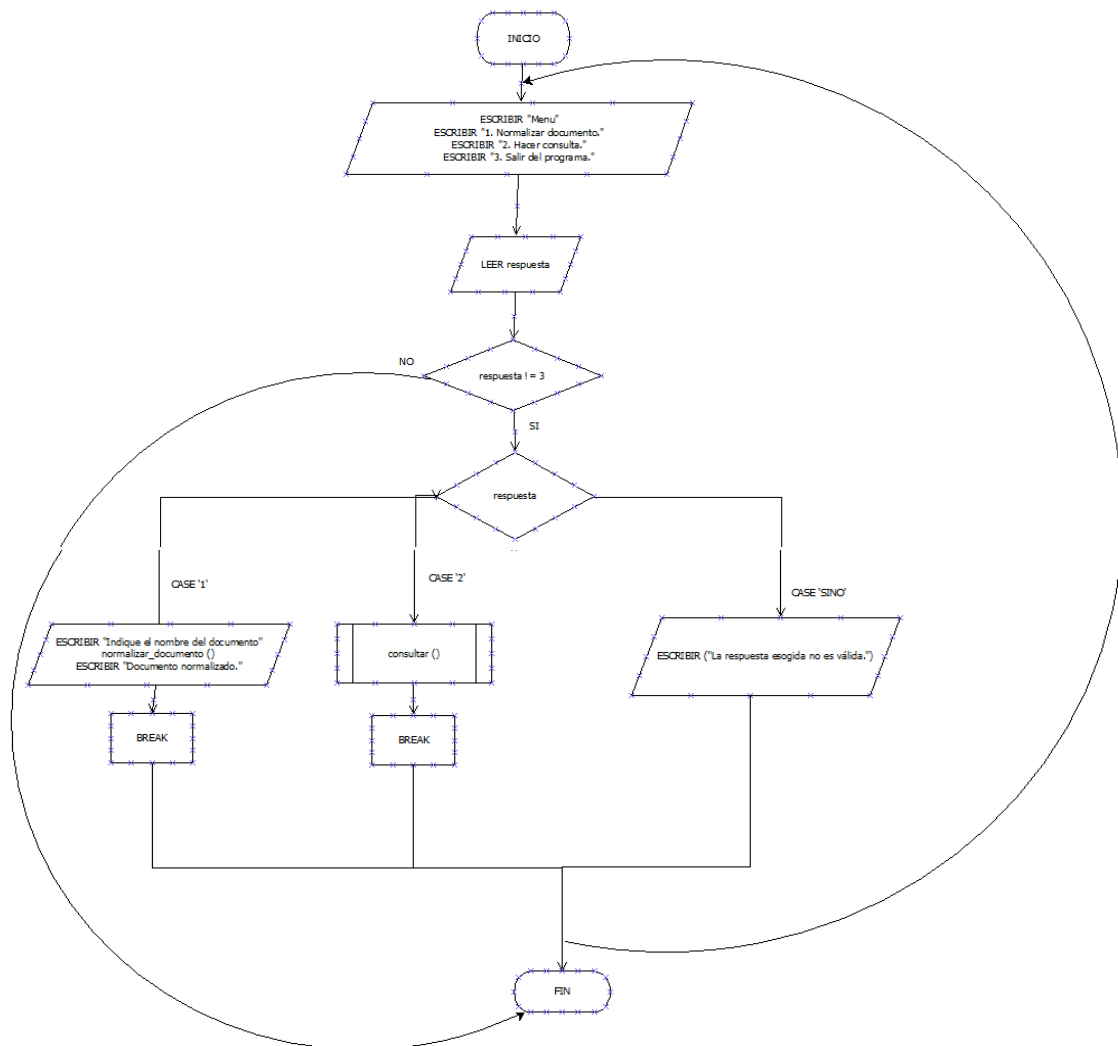
Para hacer que el programa lleve la consulta a cabo se han creado un par de funciones que leen la consulta y la dividen en tokens, y que buscan en los documentos los términos de la consulta.

2. Desarrollo

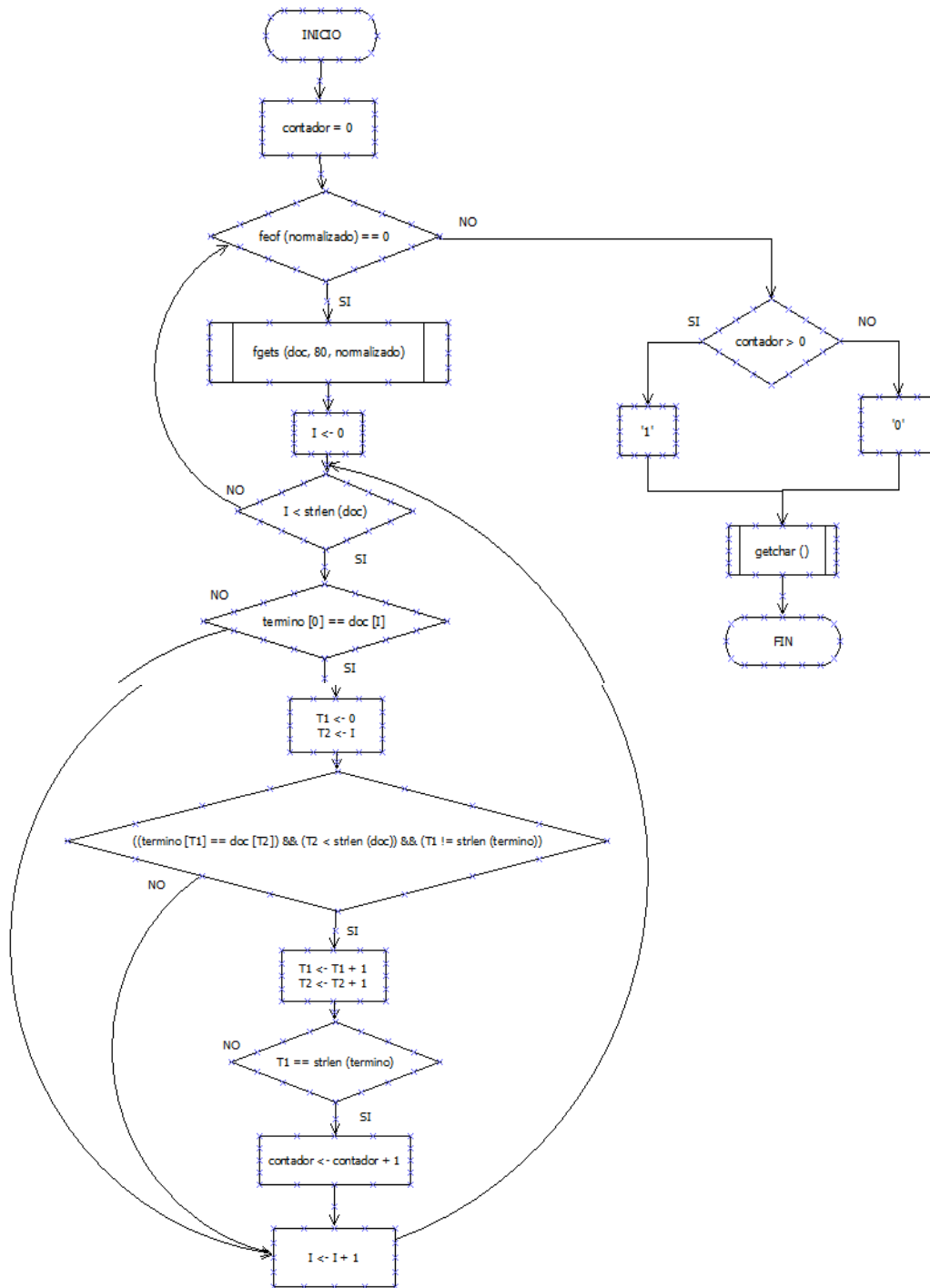
Función que lee la consulta introducida por teclado.



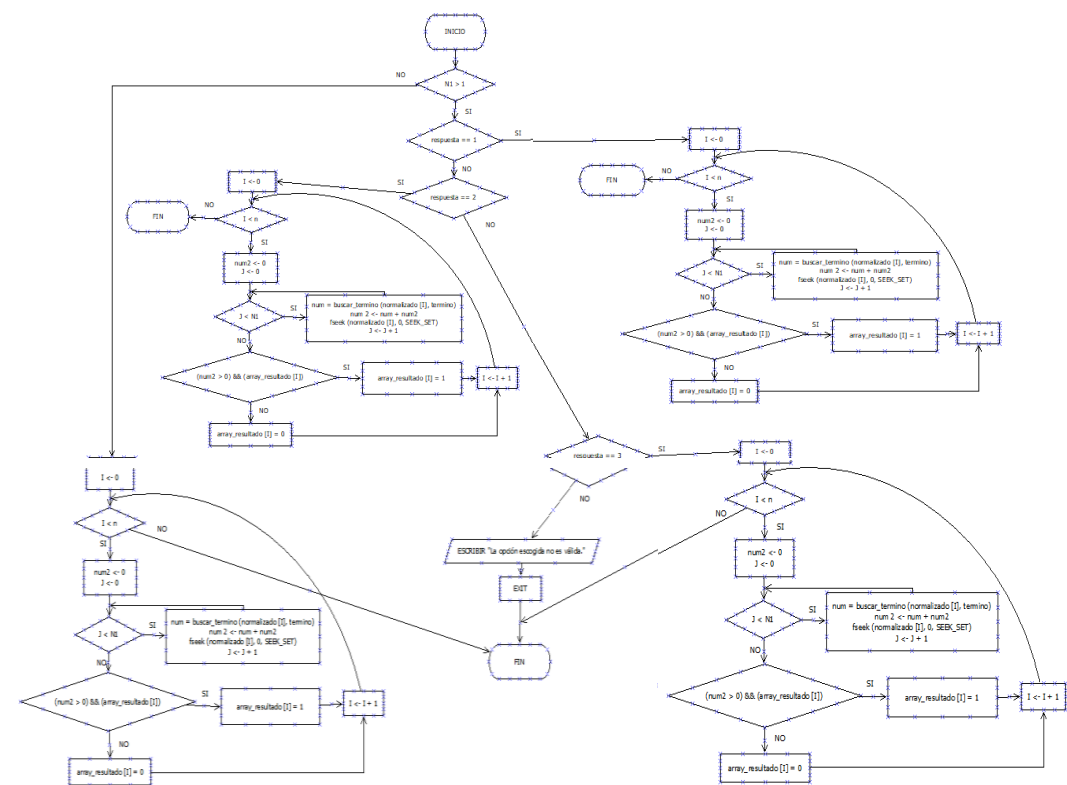
Función que muestra un menú con distintas opciones y que lee la respuesta introducida por teclado (mostrar_menu ()).



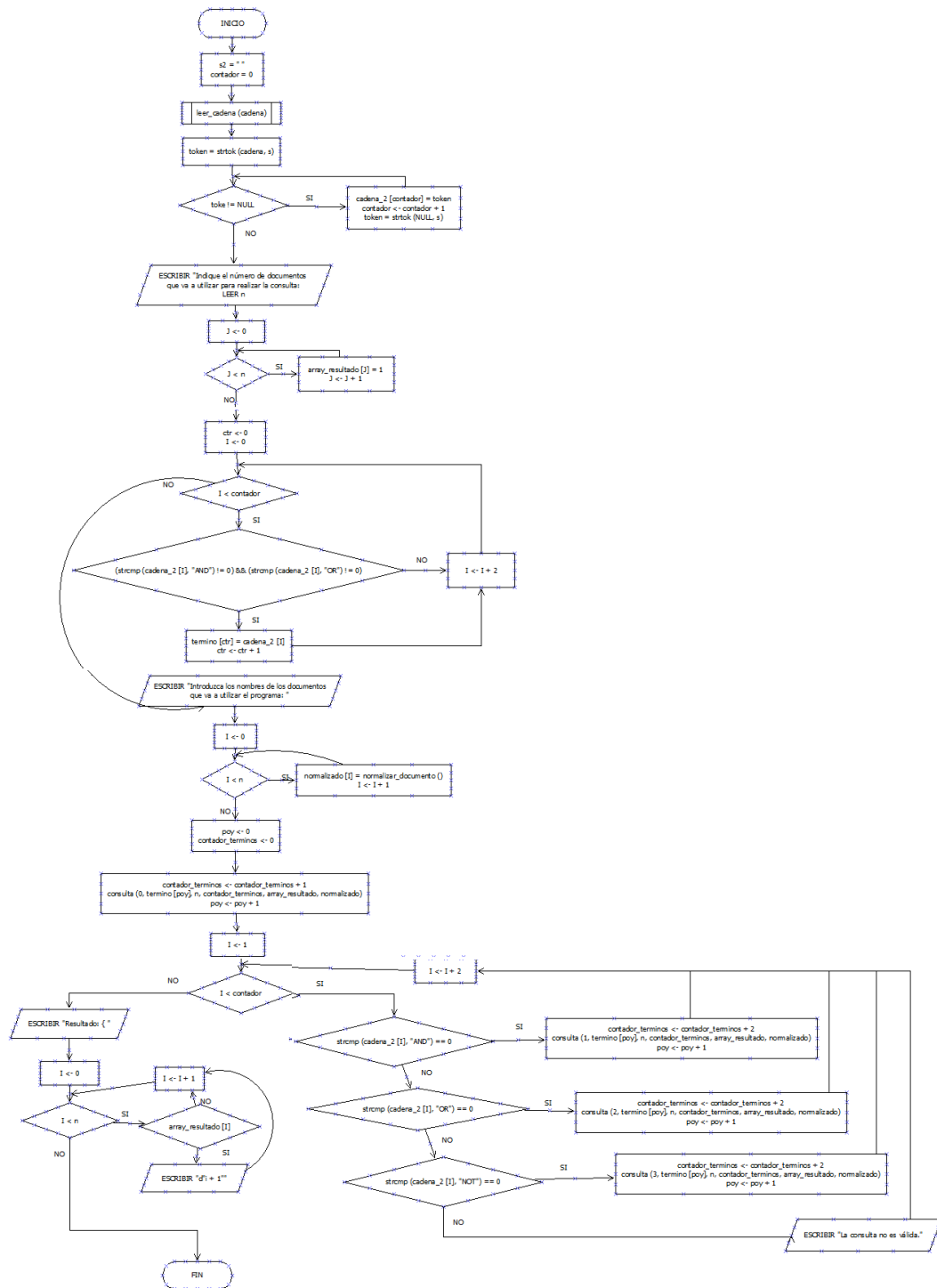
Función que busca un término introducido por parámetro en un documento (buscar_termino).



Módulo que hace una consulta



Módulo que divide la consulta leída por teclado en tokens, y mira a ver si es un AND, OR o NOT para poder hacer la consulta.



3. Pruebas

Las siguientes capturas de pantalla muestran las diferentes opciones que el programa creado

En esta primera captura de pantalla se puede ver como al escoger la opción 1 del programa normaliza el documento introducido.

```
victor@Algoritmia:~/Escritorio$ gcc Practica2RI.c
victor@Algoritmia:~/Escritorio$ ./a.out

Menu
1. Normalizar documento.
2. Hacer consulta.
3. Salir del programa.
Respuesta: 1

Indique el nombre del documento
documento1.txt

Documento normalizado.
```

En estas tres capturas se muestra como la consulta puede tener los términos que se quieran, introduciéndole el número de documentos, y el nombre de cada uno de ellos, para poder hacer la consulta.

```
Menu
1. Normalizar documento.
2. Hacer consulta.
3. Salir del programa.
Respuesta: 2

Introduzca la consulta: barco

Indique el número de documentos que va a utilizar para realizar la consulta: 4

Introduzca los nombres de los documentos que va a utilizar el programa:
documento1.txt
documento2.txt
documento3.txt
documento4.txt

Resultado: { d1 d4 }
```



```
Menu
1. Normalizar documento.
2. Hacer consulta.
3. Salir del programa.
Respuesta: 2

Introduzca la consulta: barco AND casa

Indique el número de documentos que va a utilizar para realizar la consulta: 4

Introduzca los nombres de los documentos que va a utilizar el programa:
documento1.txt
documento2.txt
documento3.txt
documento4.txt

Resultado: { d4 }
```

```
Menu
1. Normalizar documento.
2. Hacer consulta.
3. Salir del programa.
Respuesta: 2

Introduzca la consulta: barco AND casa OR coche

Indique el número de documentos que va a utilizar para realizar la consulta: 4

Introduzca los nombres de los documentos que va a utilizar el programa:
documento1.txt
documento2.txt
documento3.txt
documento4.txt

Resultado: { d2 d3 d4 }
```

Y en esta última captura de pantalla se muestra la tercera opción, que es la de salir del programa.

```
Menu
1. Normalizar documento.
2. Hacer consulta.
3. Salir del programa.
Respuesta: 3
victor@Algoritmia:~/Escritorio$
```

4. Conclusiones y observaciones

A la hora de hacer la consulta la opción AND NOT hay que escribir NOT.

Ha sido algo complicado el programa a la hora de coger la consulta, porque he tenido que dividirla en tokens, y luego mirar si era un término o una opción como AND, OR o NOT.

Pero al final, en general, ha sido algo menos complicado que el primero.

Se ha reutilizado casi todo el código de la primera práctica. Ha habido que cambiar un par de cosas al módulo `normalizar_documento()`, para que en vez de devolver nada devolviera el fichero normalizado para poder utilizarlo luego en otra función.

Al final con estas prácticas además de seguir la asignatura, aprendes C y te pones a pensar en cómo hacer un programa de forma autodidacta.

5. Bibliografía

[1] "C library function tolower()"

https://www.tutorialspoint.com/c_standard_library/c_function_toupper.htm

[2] "Dia" <http://dia-installer.de/index.html.es>

[3] "C | Mi Word Press" <https://pasky.wordpress.com/category/programacion/c/>