

Fundamentos de la programación

Curso 2013–2014

Control de abril

Tiempo disponible: 1 hora y 30 minutos

Construye un programa en C++ que encuentre todas las apariciones de una palabra en una sopa de letras.

El programa comenzará cargando la sopa de letras desde el archivo `sopa.txt`, que está estructurado en 10 líneas, cada una con 10 caracteres. Por ello, la sopa de letras resultante será una matriz de 10×10 caracteres. Usa la constante $N=10$ a lo largo de tu programa. A continuación se mostrará por pantalla la sopa, y se pedirá por pantalla la palabra a buscar. Esta palabra, que denominaremos *patrón* en lo sucesivo, se leerá desde el teclado a través de una variable del tipo `string`. Como resultado el programa mostrará el número de apariciones del patrón en la sopa. A la hora de superponer el patrón sobre la sopa, sólo se considerarán las direcciones $\rightarrow$ y $\nearrow$. Por ejemplo, al buscar el patrón "rock" en la siguiente sopa sólo se encontrarían las dos apariciones que se muestran en rojo (las que figuran en azul corresponden a apariciones dispuestas en direcciones no válidas que se ignoran).

	0	1	2	3	4	5	6	7	8	9
0	a	b	c	d	e	k	g	h	i	j
1	a	b	c	d	e	c	r	o	c	k
2	a	b	c	d	e	o	g	h	i	j
3	a	b	c	d	c	r	g	h	i	j
4	a	b	c	k	e	f	g	h	i	j
5	a	b	c	d	e	f	g	h	i	k
6	a	b	c	d	e	f	g	h	c	j
7	k	c	o	r	e	f	g	o	i	j
8	a	b	c	d	e	f	r	h	i	j
9	a	b	c	d	e	f	g	h	i	j

Debes comenzar definiendo el tipo `tSopa`. Luego, debes diseñar e implementar, al menos, funciones para:

- A. [1] Cargar la sopa desde el fichero.
 - B. [0,5] Mostrar la sopa.
 - C. [2] Determinar si el patrón se encuentra en la sopa a partir de una casilla dada en la dirección $\rightarrow$.
 - D. [2] Determinar si el patrón se encuentra en la sopa a partir de una casilla dada en la dirección $\nearrow$.
 - E. [2] Contar todas las apariciones del patrón en la sopa. Se trata de probar en cada casilla si el patrón puede superponerse sobre la sopa a partir de ella. Como ya se ha dicho, sólo se consideran las direcciones $\rightarrow$ y $\nearrow$.
- [0,5] Uso de constantes, declaración de tipos y la función `main`.

Debes implementar las búsquedas y los recorridos con los esquemas vistos en clase. Toda búsqueda debe concluir en cuanto se encuentre lo que se busca. Cuida el estilo. No se permiten bucles anidados de forma explícita, salvo bucles `for`.

A continuación queremos introducir colores en nuestro programa. Usa una máscara para indicar qué casillas de la sopa están marcadas como pertenecientes a una aparición del patrón. Al principio ninguna casilla de la sopa estará marcada. Luego, cuando se encuentre una nueva aparición, deben marcarse en la máscara todas las casillas pertenecientes a ella. Para desarrollar esta extensión debes:

- F. [1] Modificar la función del apartado E para que actualice la máscara registrando las apariciones encontradas.
- G. [0,5] Declarar el tipo `tMascara`, e implementar una función para inicializar la máscara.
- H. [0,5] Modifica la función del apartado B para mostrar las casillas pertenecientes a una aparición en rojo, y el resto en blanco.