

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

27-11-2024

SOPA DE LETRAS

Práctica 3

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

Mauro Hernández Cardenal
PROGRAMACIÓN

En esta práctica teníamos que hacer una sopa de letras y ser capaces de encontrar la palabra que queríamos.

<https://github.com/Maurohc91/SOPA-DE-LETRAS>

En primer lugar, voy a capturar el código y luego pasaremos a ver los posibles fallos y el funcionamiento.

```
1 package org.example;
2
3 import java.sql.SQLOutput;
4 import java.util.Scanner;
5
6 public class Main {
7     public static void main(String[] args) {
8
9         Scanner teclado = new Scanner(System.in);
10
11         //demanelem a l'usuari el tamany de la matriu
12
13         System.out.println("Introduce el número de filas (m): ");
14         int m = teclado.nextInt(); //numero de files
15         System.out.println("Introduce el número de columnas (n): ");
16         int n = teclado.nextInt(); //numero de columnas
17
18         String matriz [][] = new String[m][n];
19
20         //inicialitzem la matriu
21         for (int i = 0; i < m; i++) {
22             System.out.println("Introduce fila " + (i+1) + " debe contener " + n + " letras.");
23             String fila = teclado.next();
24
25             //validem que la fila continga exactament les lletres correctes i que siguin lletres
26             while (!fila.matches(regex: "[a-zA-Z]{n}")){
27                 System.out.println("ERROR. Vuelve a introducir de nuevo.");
28                 fila = teclado.next();
29             }
30
31             //plenem la fila en la matriu
32             for (int j = 0; j < n; j++) {
33                 matriz[i][j] = String.valueOf(fila.charAt(j));
34             }
35         }
36         //mostrem la matriu
37
38         System.out.print("La sopa de letras es la siguiente: ");
39         System.out.println();
40         for (String[] filas : matriz){
41             for (String columnas : filas) {
42                 System.out.print(columnas + " ");
43             }
44             System.out.println();
45         }
46         //demanelem la paraula a buscar
```

```

47     System.out.println("Introduce la palabra a buscar:");
48     String palabra = teclado.next();
49
50     //per a veure que la paraula només conte lletres
51
52     while (!palabra.matches( regex: "[a-zA-Z]+")) {
53         System.out.println("ERROR. Vuelva a introducir la palabra: ");
54         palabra = teclado.next();
55     }
56
57     //ara buscarem la paraula en la matriu
58     boolean encontrada = false;
59
60     for (int i = 0; i < m && !encontrada; i++) {
61         for (int j = 0; j < n && !encontrada; j++) {
62             //per a coincidencia horitzontal
63             boolean horizontal = true;
64             if (j + palabra.length() <= n) {
65                 for (int k = 0; k < palabra.length(); k++) {
66                     if (!matriz[i][j+k].equalsIgnoreCase(String.valueOf(palabra.charAt(k)))) {
67                         horizontal = false;
68                         break;
69                     }
70                 }

```

```

71             }else{
72                 horizontal=false;
73             }
74
75             //per a coincidencia vertical
76             boolean vertical = true;
77             if (i + palabra.length() <=m) {
78                 for (int k = 0; k < palabra.length(); k++) {
79                     if (!matriz [i+k][j].equalsIgnoreCase(String.valueOf(palabra.charAt(k)))) {
80                         vertical = false;
81                         break;
82                     }
83                 }
84             }else {
85                 vertical=false;
86             }
87             //per a veure si la paraula ha sigut encontrada
88             if (horizontal || vertical) {
89                 System.out.println("PALABRA ENCONTRADA. POSICION: " + i + ", " + j);
90                 encontrada=true;
91             }
92         }
93     }

```

```

93     }
94     if (!encontrada) {
95         System.out.println("La palabra no existe.");
96     }
97
98 }
99

```

Y ahora, vamos a ver los posibles fallos.

Por ejemplo cuando metemos un número y no una letra.

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaag
Introduce el número de filas (m):
4
Introduce el número de columnas (n):
4
Introduce fila 1 debe contener 4 letras.
ñlk2
ERROR. Vuelve a introducir de nuevo.
|
```

Cuando pones más letras de las que pides.

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-ja
Introduce el número de filas (m):
4
Introduce el número de columnas (n):
4
Introduce fila 1 debe contener 4 letras.
asdfff
ERROR. Vuelve a introducir de nuevo.
|
```

Cuando pones un símbolo.

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-j
Introduce el número de filas (m):
4
Introduce el número de columnas (n):
4
Introduce fila 1 debe contener 4 letras.
asd/
ERROR. Vuelve a introducir de nuevo.
|
```

Y aquí la captura con el funcionamiento correcto del programa.

```
"C:\Program Files\Java\jdk-23\bin\java.exe" "-javaagent:
Introduce el número de filas (m):
4
Introduce el número de columnas (n):
4
Introduce fila 1 debe contener 4 letras.
aklj
Introduce fila 2 debe contener 4 letras.
mklk
Introduce fila 3 debe contener 4 letras.
okok
Introduce fila 4 debe contener 4 letras.
roik
La sopa de letras es la siguiente:
a k l j
m k l k
o k o k
r o i k
Introduce la palabra a buscar:
amor
PALABRA ENCONTRADA. POSICION: 0,0

Process finished with exit code 0
|
```