



# Entorns de Desenvolupament

## Sopa de lletres exercici 3




AVESTRUZ


O	C	A	M	E	L	L	O	K	M
S	T	V	I	N	T	W	S	U	P
C	I	E	Y	B	U	I	T	R	E
A	F	S	H	E	J	I	K	O	M
O	U	T	A	Q	P	E	R	Z	I
Ñ	W	R	B	T	C	O	B	R	A
O	L	U	G	U	C	F	A	N	Y
H	E	Z	S	G	A	C	E	L	A
I	V	D	O	J	X	U	K	L	A
G	E	C	A	M	A	L	E	O	N




CAMELLO




BUITRE



COBRA



CAMALEÓN



GACELA

Alumnos: José María Lagreca, Daniel Sánchez

Profesora: José Luís Luján



## Índex

<b>Enunciat.....</b>	<b>2</b>
<b>Solució.....</b>	<b>6</b>
Obtenir lletra aleatòria.....	6
Acció per emplenar amb lletres aleatòries.....	6
- Sortida de la sopa de lletra.....	6
Paraula a la sopa de lletres.....	7

## Enunciat

Continuant amb l'aplicació de la "sopa de lletres", el que se'ns demana ara és que dissenyem un algorisme que sigui capaç d'omplir una quadrícula amb lletres aleatòries i, a continuació, hi insereixi les diferents paraules que caldrà cercar després. Les paraules sobreescriran les lletres, per la qual cosa, el resultat final serà el de una "sopa de lletres" (les paraules estaran camuflades entre les lletres prèviament escrites).

Tenim la següent seqüència d'entrada on apareix la informació que necessitem:

```
< N M W word1 word2 word3 ... wordW >
```

On:

- N i M les dimensions de la quadrícula de la sopa de lletres: N files x M columnes.
- W el nombre de paraules a afegir a la sopa de lletres.
- wordi la paraula i-èsima que cal afegir a la sopa de lletres, en majúscules.

El següent algorisme llegeix la seqüència d'entrada, crea una quadrícula de N x M, l'omple de lletres aleatòriament i, per últim, hi sobreesciu les paraules situant-les al atzar sobre la quadrícula. El resultat es mostra en una seqüència de sortida amb el següent format:

```
< N M c1 c2 c3 c4 ... cNxM >
```

On:

N i M són les mateixes dimensions de la quadrícula que hem llegit de l'entrada.

ci és el caràcter i-èsim de la sopa de lletres, **en majúscules**.

Es demana que completeu els **punts suspensius** que falten per tal que l'algorisme es comporti tal com s'ha descrit.

**NOTA:** Utilitzeu les estructures de dades, accions i funcions declarades en els exercicis anteriors quan sigui necessari. També podeu utilitzar la funció següent, que us proporciona un nombre aleatori entre dos enters donats:

**funcio** getRandomNumber( min: **enter**, max: **enter** ): **enter**

**algorisme** fillSoup

**var**

soup: **tSoup**;  
search: **tSearch**;  
word: **tWord**;  
n,m,w,i: **enter**;

**fvar**

```

n:= readInteger();
m:= readInteger();
initSoup( n,m, soup );
initSearch( search );

fillWithRandomCharacters( n, m, soup);
w:= readInteger();
per i:= 1 fins w fer
    word:= .....
    addWordToSearch( search, word );
    hideWordIntoSoup(word, n, m, soup);
fper
writeSoup( n, m, soup);

```

**algoritme**

**funcio** getRandomCharacter(): **caracter**

```

var
    min, max, code: enter;
fvar
    .....

```

**funcio**

**accio** fillWithRandomCharacters(**ent** n: **enter**, **ent** m: **enter**, **entsor** soup: **tSoup**)

```

var
    c: caracter;
    i,j: enter;
fvar
per i:= 1 fins n fer
    per j:= 1 fins m fer
        c:= getRandomCharacter();

```

.....  
**fper**

**fper**

**faccio**

**accio** writeSoup(**ent** n,m: **enter**, **ent** soup: **tSoup**)

```

var
    i,j: enter;
fvar

```

```

per i:= 1 fins n fer
    per j:= 1 fins m fer
        writeChar( ..... );
    fper
fper
faccio

```

```

accio hideWordIntoSoup( entsor word: tWord, ent n: enter, ent m: enter, entsor soup: tSoup
)

```

```

    var
        i,j,ori,len: enter;
    fvar
        len:= getWordLength(word); ori:= getRandomNumber(1,3);
        si ori = 1 llavors { case: VERTICAL }
            i:=getRandomNumber(1,n-len+1);
            j:=getRandomNumber(1,m);
            .....
        sino
            si ori = 2 llavors { case: HORIZONTAL }
                i:=getRandomNumber(1,n);
                j:=getRandomNumber(1,m-len+1);
            sino { case: DIAGONAL }
                .....
            fsi
        fsi
faccio

```

```

accio addWordToSearch( entsor search: tSearch, ent word: tWord )

```

```

    .....
faccio

```

```

accio initSearch( sor search: tSearch )

```

```

    .....
faccio

```

```

accio initSoup( ent n,m: enter, sor soup: tSoup )

```

```

    .....

```

  
**facció**

## Solució

### Obtenir lletra aleatòria

```
funcio getRandomCharacter(): caracter
var
    min, max, code: enter;
fvar
    min := 65;
    max := 90;
    code := getRandomNumber(min, max);
    retorna chr(code);
ffuncio
```

### Acció per emplenar amb lletres aleatòries

```
accio fillWithRandomCharacters(ent n: enter, ent m: enter, entsor soup: tSoup)
var
    c: caracter;
    i, j: enter;
fvar
    per i := 1 fins n fer
        per j := 1 fins m fer
            c := getRandomCharacter();
            setSoupChar(i, j, c, soup);
        fper
    fper
faccio
```

### - Sortida de la sopa de lletra

```
accio writeSoup(ent n,m: enter, ent soup: tSoup)
var
    i, j: enter;
    c: caracter;
```

```

fvar
  write(n); write(' '); write(m);
per i := 1 fins n fer
  per j := 1 fins m fer
    c := getSoupChar(i, j, soup);
    write(' ');
    writeChar(c);
  fper
fper
faccio

```

## Paraula a la sopa de lletres

```

accio hideWordIntoSoup(entsor word: tWord, ent n: enter, ent m: enter, entsor soup: tSoup)
var
  i, j, k, ori, len: enter;
  c: caracter;
fvar
  len := getWordLength(word);
  ori := getRandomNumber(1, 3);

  si ori = 1 llavors
    i := getRandomNumber(1, n - len + 1);
    j := getRandomNumber(1, m);
    per k := 1 fins len fer
      c := getCharacterAt(word, k);
      setSoupChar(i + k - 1, j, c, soup);
    fper
  sino si ori = 2 llavors
    i := getRandomNumber(1, n);
    j := getRandomNumber(1, m - len + 1);
    per k := 1 fins len fer
      c := getCharacterAt(word, k);
      setSoupChar(i, j + k - 1, c, soup);
    fper
  sino
    i := getRandomNumber(1, n - len + 1);
    j := getRandomNumber(1, m - len + 1);
    per k := 1 fins len fer
      c := getCharacterAt(word, k);

```



```
    setSoupChar(i + k - 1, j + k - 1, c, soup);  
    fper  
    fsi  
faccio
```