

UF1 – PROGRAMACIÓ ESTRUCTURADA

A05

ARRAYS BIDIMENSIONALS

Introducció

Never trust to general impressions, my boy, but concentrate yourself upon details.

—SIR ARTHUR CONAN DOYLE, A CASE OF IDENTITY
(SHERLOCK HOLMES)

	0	1	2
0	✗	●	●
1	●	✗	
2			✗

Introducció a les matrius bidimensionals

- Per a poder accedir a una dada de la matriu s'han d'indicar **dues posicions** i, per tant, dos índexs.

	Columna 0	Columna 1	Columna 2	Columna 3
Fila 0	A[0][0]	A[0][1]	A[0][2]	A[0][3]
Fila 1	A[1][0]	A[1][1]	A[1][2]	A[1][3]
Fila 2	A[2][0]	A[2][1]	A[2][2]	A[2][3]

Subíndice de columna
Subíndice de fila
Nombre de arreglo

- Es una manera convenient d'emmagatzemar informació en una taula rectangular i referir-se a ella a través de les seves **files** i **columnes**.

Exemple de matriu bidimensional

```
for (int i = 0; i < M; i++)
{
    for (int j = 0; j < N; j++)
        System.out.print(a[i][j] + " ")
    System.out.println();
}
```

Fila Columna



a[0][0]	a[0][1]	a[0][2]
a[1][0]	a[1][1]	a[1][2]
a[2][0]	a[2][1]	a[2][2]
a[3][0]	a[3][1]	a[3][2]
a[4][0]	a[4][1]	a[4][2]
a[5][0]	a[5][1]	a[5][2]
a[6][0]	a[6][1]	a[6][2]
a[7][0]	a[7][1]	a[7][2]
a[8][0]	a[8][1]	a[8][2]
a[9][0]	a[9][1]	a[9][2]

Fem ús de *nested loops*:

Per cada fila **i** recorrem totes les columnes **j**

En aquest exemple tenim una **matriu de 10-per-3** (10 files i 3 columnes)

Matrius numèriques bidimensionals

- Per a crear i utilitzar una matriu s'ha de realitzar tres operacions:

- 1. Declarar-la**
- 2. Crear-la**
- 3. Inicialitzar-la**

Declarar una matriu bidimensional

- La declaració en Java d'una matriu de dues dimensions es fa de dues formes diferents:

tipus[][] nom;

tipus nom[][];

Int [] [] numeros;	Declara un array bidimensional de tipus int anomenat <i>numeros</i>
String [] [] noms;	Declara un array bidimensional anomenat noms que contindrà elements de tipus String

Crear una matriu bidimensional

- Els *arrays bidimensionals* es creen amb l'operador new tal com es fa amb les arrays unidimensionals.

nom = new tipus[NUM_FILES][NUM_COLUMNS];

```
numeros = new  
int[10][5];
```

La primera dimensió especifica que l'array numeros té 10 elements. La segona dimensió especifica que cada element anterior és a sí mateix un array de 5 elements.

Iniciar una matriu bidimensional

- **Una matriu és un objecte** i, per tant, els seus elements són automàticament inicialitzats.
- Si es desitja **iniciar una matriu bidimensional** amb altres valors diferents als predeterminats, es pot fer de la següent manera:

```
int [][] temperatura = {{1, 2, 3}, {4, 5, 6}};
```

Java determina la grandària de l'array en funció dels valors assignats i fa la reserva de memòria sense haver de fer el
new

Accedir als elements d'una matriu

- Per **accedir al valor d'un element d'una matriu bidimensional** s'ha d'indicar dues posicions; per això, s'anomenen taules bidimensionals.
- A través de les dues posicions se li indica a quina fila i a quina columna, hi ha la dada a la que es vol accedir.

matriu[posFila][posColumna]

Accedir als elements d'una matriu

- Java comença a enumerar les files pel 0, igual que les columnes. Així doncs, la `matriu[0][0]` es refereix al primer element de la matriu.
- Per a saber quin és la grandària d'una taula bidimensional es fa servir la propietat `length`:
 - **`matriu.length`** ens donarà el número de files
 - **`matriu[posF].length`** ens donarà el número de columnes que té la fila **`posF`**.

Recorregut

- Per a recórrer una matriu bidimensional, són necessàries dues estructures repetitives: una per a recórrer cada una de les files de l'array i l'altra per a recorre cada una de les columnes.

```
for(int posF = 0; posF < matriu.length; posF++) {  
    ...  
    for(int posC = 0; posC < matriu[posF].length;  
    posC++) {  
        ...  
    }  
}
```

Recorregut: Exemple 1

```
int[][] temp = {{1,2,3},{4,5,6}};
```

```
for(int f=0; f<temp.length; f++){
```

```
    // Codi a executar abans de cada fila
```

```
    for(int c=0; c<temp[f].length; c++){
```

```
        // Codi a executar per a cada cel·la [f,c]
```

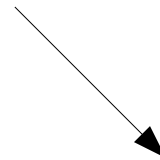
```
        System.out.print(temp[f][c] + " ");
```

```
    }
```

```
    // Codi a executar després de cada fila
```

```
    System.out.println("");
```

```
}
```

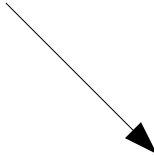


1	2	3
4	5	6

Recorregut: Exemple 2

```
String [][] months = { {"Jan","Feb","Mar"}, {"Apr","May","Jun"},  
                        {"Jul","Aug","Sep"}, {"Oct","Nov","Des"} };
```

```
for(int f=0; f<months.length; f++){  
    // Codi a executar abans de cada fila  
    for(int c=0; c<months[f].length; c++){  
        // Codi a executar per a cada cel·la [f,c]  
        System.out.print(months[f][c] + " ");  
    }  
    // Codi a executar després de cada fila  
    System.out.println("");  
}
```



Jan	Feb	Mar
Apr	May	Jun
Jul	Aug	Sep
Oct	Nov	Des