

Taller de preparación para la OCI - PUC

## Guía de Ejercicios Semana 11

### Introducción

En esta guía encontrarás problemas de arreglos multidimensionales.

### Ejercicios

1. Escribe una función:

```
1 public static void imprimir(int [][] arreglo_bidimensional) {
2     // ...
3 }
```

Que recibe un arreglo de dos dimensiones y lo imprime en consola.

2. Escribe una función:

```
1 public static int sumar_todo(int [][] arreglo_bidimensional, int n) {
2     // ...
3 }
```

Que recibe un arreglo de dos dimensiones y un número  $n$ . La función debe retornar la suma de todos los elementos que están el arreglo de dos dimensiones y son mayores a  $n$ .

3. En este programa vamos a simular una lista de compra. Tu programa va a recibir un número  $n$ . Luego vendrán  $n$  líneas que representan productos junto a su precio por unidad (separados por espacio). Luego vendrá un número  $m$  tal que ( $m \leq n$ ). Ahí vendrán el nombre de los productos a comprar y su cantidad. Tu programa debe imprimir el total de la compra.

#### Ejemplo 1:

```
5
Papa 300
Queso 200
Tomate 100
Carne 500
Bebida 70
3 Papa 3
Queso 1
Carne 2
El total de la compra es: $2100
```

4. Escribe un programa que reciba un número  $n$ . Luego vendrán  $n$  líneas, que representa un listado de alumnos y sus calificaciones (los elementos vienen separados por espacios como se ve en el ejemplo). Tu programa debe imprimir cada alumno junto a su promedio. Finalmente se debe imprimir el promedio del curso y la cantidad de alumnos aprobados (promedio mayor o igual a 4.0) y reprobados (promedio menor que 4.0).

Entrada	Salida
3	
Pedro 2.0 3.0 4.0	Pedro 3.0
Juan 4.5 5.0 5.5	Juan 5.0
Diego 1.0 4.0 7.0	Diego 4.0
	Promedio Curso: 4.0
	Aprobados: 2 - Reprobados: 1

5. Escribe un programa que simule el popular juego “Gato” ([https://es.wikipedia.org/wiki/Tres\\_en\\_l%C3%ADnea](https://es.wikipedia.org/wiki/Tres_en_l%C3%ADnea)). Tu programa deberá pedir en cada turno la coordenada  $(i, j)$  que representa el movimiento. El jugador que parte es la “X”. Cuando ya no queden más jugadas posibles, tu programa debe decidir quien fue el ganador. **Observación:** No se pueden hacer jugadas no válidas, como ingresar una posición no válida o que ya esté ocupada. El sistema de coordenadas queda a criterio del programador.
6. Un cuadrado mágico es un tablero cuadrado con números en el que la suma de cada una de sus filas, de cada una de sus columnas y de cada una de sus diagonales tienen el mismo valor. En este ejercicio debes hacer un programa que reciba un número  $n$  que representa el número de filas y columnas. Luego vendrán las  $n$  filas del cuadrado, que son números separados por espacio. Tu programa debe imprimir si el input corresponde a un cuadrado mágico.

#### Ejemplo 1:

```
3
1 2 3
4 5 6
7 8 9
```

Respuesta: No es un cuadrado mágico

#### Ejemplo 2:

```
5
17 24 1 8 15
23 5 7 14 16
4 6 13 20 22
10 12 19 21 3
11 18 25 2 9
```

Respuesta: Es un cuadrado mágico

7. En este ejercicio vamos a trabajar un poco la lógica que gobierna el clásico juego Bomberman. En este juego hay que ir colocando bombas para derrotar a tus oponentes. Las bombas demoran un par de segundos en explotar y son colocadas dentro de un sistema de cuadrícula. De este modo la explosión solo tienen un efecto horizontal y vertical (siguiendo la cuadrícula).



Figura 1: Típica secuencia Super Bomberman 5 de Super Nintendo (SNES). Para más referencias pueden buscar “Bomberman 5 SNES” en Youtube.

Su programa recibe un juego y debe decidir cuáles jugadores sobreviven y cuáles mueren luego de que explote la bomba. En la primera línea usted recibirá el ancho del tablero, el alto del tablero y el poder de la bomba. El poder de la bomba es simplemente cuantos espacios verticales y horizontales tiene de alcance la bomba. La bomba, al explotar, acaba con todos los jugadores dentro del alcance que estén de forma vertical u horizontal. La explosión no puede traspasar paredes.

Luego, se le entregará el tablero como una secuencia de caracteres. En el tablero los caracteres “O” son espacios libres, las “X” son paredes, las “A”, “B”, “C” y “D” son jugadores, y “N” es la bomba. Puede suponer que solo habrá una bomba y que la explosión ocurre instantáneamente (los jugadores no se pueden mover).

#### Ejemplo 1:

```

9 9 3
O O O O O O O O O
O X O X O X O X O
O O O O A O O O O
O X O X O X O X O
B O O O N D O O O
O X O X O X O X O
O O O C O O O O O
O X O X O X O X O
O O O O O O O O O
A muere
B sobrevive
C sobrevive

```

D muere

### Ejemplo 2:

9 9 6

```
0 0 0 A 0 0 0 0 N
0 X 0 X 0 X 0 X 0
0 0 0 0 0 0 0 0 B
0 X 0 X 0 X 0 X 0
0 0 0 0 0 0 0 0 0
0 X 0 X 0 X 0 X 0
0 0 D 0 0 0 0 0 0
0 X 0 X 0 X 0 X C
0 0 0 0 0 0 0 0 0
```

A muere

B muere

C sobrevive

D sobrevive

### Ejemplo 3:

9 9 2

```
0 0 0 0 0 0 0 0 0
0 X 0 X 0 X 0 X 0
0 0 0 0 0 0 B 0 0
0 X 0 X 0 X X X 0
0 0 0 0 A X N 0 C
0 X 0 X 0 X D X 0
0 0 0 0 0 0 0 0 0
0 X 0 X 0 X 0 X 0
0 0 0 0 0 0 0 0 0
```

A sobrevive

B sobrevive

C muere

D muere

### Ejemplo 4:

9 9 4

```
0 0 0 0 0 0 0 0 0
0 X 0 X B X 0 X 0
0 0 0 0 A 0 0 0 0
0 X 0 X X X 0 X 0
0 0 C X N X 0 0 0
0 X 0 X X X 0 X 0
0 0 0 0 D 0 0 0 0
0 X 0 X 0 X 0 X 0
0 0 0 0 0 0 0 0 0
```

A sobrevive

B sobrevive

C sobrevive

D sobrevive