



# EJERCICIOS ARRAYS UNIDIMENSIONALES Y MULTIDIMENSIONELS

## PROGRAMACIÓN

10 EJERCICIOS MÍNIMOS PARA PODER SACAR UN 5

## EJERCICIOS PARA OBTENER UN 5

### ARRAYS UNIDIMENSIONALES

#### Ejercicio 1



Define un *array* de 12 números enteros con nombre `num` y asigna los valores según la tabla que se muestra a continuación. Muestra el contenido de todos los elementos del *array*. ¿Qué sucede con los valores de los elementos que no han sido inicializados?

Índice	0	1	2	3	4	5	6	7	8	9	10	11
Valor	39	-2			0		14		5	120		

#### Ejercicio 2

Escribe un programa que lea 10 números por teclado y que luego los muestre en orden inverso, es decir, el primero que se introduce es el último en mostrarse y viceversa.

#### Ejercicio 3

Define tres *arrays* de 20 números enteros cada una, con nombres `numero`, `cuadrado` y `cubo`. Carga el *array* `numero` con valores aleatorios entre 0 y 100. En el *array* `cuadrado` se deben almacenar los cuadrados de los valores que hay en el *array* `numero`. En el *array* `cubo` se deben almacenar los cubos de los valores que hay en `numero`. A continuación, muestra el contenido de los tres *arrays* dispuesto en tres columnas.

#### Ejercicio 4

Escribe un programa que lea 15 números por teclado y que los almacene en un *array*. Rota los elementos de ese *array*, es decir, el elemento de la posición 0 debe pasar a la posición 1, el de la 1 a la 2, etc. El número que se encuentra en la última posición debe pasar a la posición 0. Finalmente, muestra el contenido del *array*.

#### Ejercicio 5

Realiza un programa que pida la temperatura media que ha hecho en cada mes de un determinado año y que muestre a continuación un diagrama de barras horizontales con esos datos. Las barras del diagrama se pueden dibujar a base

de asteriscos o cualquier otro carácter.

## ARRAYS MULTIDIMENSIONALES

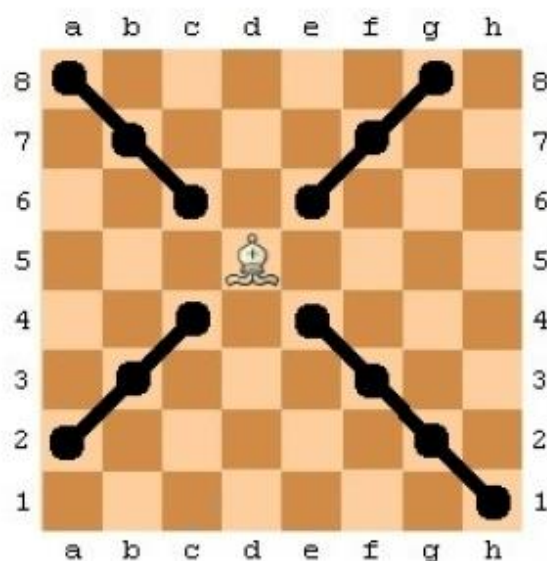
### Ejercicio 6

Define un *array* de números enteros de 3 filas por 6 columnas con nombre `num` y asigna los valores según la siguiente tabla. Muestra el contenido de todos los elementos del *array* dispuestos en forma de tabla como se muestra en la figura.

Array num	Columna 0	Columna 1	Columna 2	Columna 3	Columna 4	Columna 5
Fila 0	0	30	2			5
Fila 1	75				0	
Fila 2			-2	9		11

### Ejercicio 7

Escribe un programa que, dada una posición en un tablero de ajedrez, nos diga a qué casillas podría saltar un alfil que se encuentra en esa posición. Como se indica en la figura, el alfil se mueve siempre en diagonal. El tablero cuenta con 64 casillas. Las columnas se indican con las letras de la "a" a la "h" y las filas se indican del 1 al 8.



Ejemplo:

Introduzca la posición del alfil: d5  
El alfil puede moverse a las siguientes posiciones:  
h1 a2 g2 b3 f3 c4 e4 c6 e6 b7 f7 a8 g8

## Ejercicio 8

Realiza el juego de las tres en raya.

## Ejercicio 9

Realiza un programa que muestre por pantalla un array de 10 filas por 10 columnas relleno con números aleatorios entre 200 y 300. A continuación, el programa debe mostrar los números de la diagonal que va desde la esquina superior izquierda a la esquina inferior derecha, así como el máximo, el mínimo y la media de los números que hay en esa diagonal.

## Ejercicio 10

Realiza un programa que muestre por pantalla un array de 9 filas por 9 columnas relleno con números aleatorios entre 500 y 900. A continuación, el programa debe mostrar los números de la diagonal que va desde la esquina inferior izquierda a la esquina superior derecha, así como el máximo, el mínimo y la media de los números que hay en esa diagonal

## EJERCICIOS PARA OBTENER UN 10

### ARRAYS UNIDIMENSIONALES

## Ejercicio 11



Define un **array** de 10 caracteres con nombre **simbolo** y asigna valores a los elementos según la tabla que se muestra a continuación. Muestra el contenido de todos los elementos del **array**. ¿Qué sucede con los valores de los elementos que no han sido inicializados?

Índice	0	1	2	3	4	5	6	7	8	9
Valor	'a'	'x'			'@'		' '	'+'	'Q'	

## Ejercicio 12

Escribe un programa que genere 100 números aleatorios del 0 al 20 y que los muestre por pantalla separados por espacios. El programa pedirá entonces por teclado dos valores y a continuación cambiará todas las ocurrencias del primer valor por el segundo en la lista generada anteriormente. Los números que se han cambiado deben aparecer entrecomillados.

### Ejercicio 13

Realiza un programa que pida 8 números enteros y que luego muestre esos números junto con la palabra “par” o “impar” según proceda.

### Ejercicio 14

Realiza un programa que pida 10 números por teclado y que los almacene en un *array*. A continuación se mostrará el contenido de ese *array* junto al índice (0 – 9). Seguidamente el programa pedirá dos posiciones a las que llamaremos “inicial” y “final”. Se debe comprobar que inicial es menor que final y que ambos números están entre 0 y 9. El programa deberá colocar el número de la posición inicial en la posición final, rotando el resto de números para que no se pierda ninguno. Al final se debe mostrar el *array* resultante.

Por ejemplo, para inicial = 3 y final = 7:

Índice	0	1	2	3	4	5	6	7	8	9
Valor	20	5	7	4	32	9	2	14	11	6

**Array inicial**

Índice	0	1	2	3	4	5	6	7	8	9
Valor	6	20	5	7	32	9	2	4	14	11

**Array final**

### Ejercicio 15

sus mesas. En una mesa se pueden sentar de 0 (mesa vacía) a 4 comensales (mesa llena). Cuando llega un cliente se le pregunta cuántos son. De momento el programa no está preparado para colocar a grupos mayores a 4, por tanto, si un cliente dice por ejemplo que son un grupo de 6, el programa dará el mensaje **“Lo siento, no admitimos grupos de 6, haga grupos de 4 personas como máximo e intente de nuevo”**. Para el grupo que llega, se busca siempre la primera mesa libre (con 0 personas). Si no quedan mesas libres, se busca donde haya un hueco para todo el grupo, por ejemplo si el grupo es de dos personas, se podrá colocar donde haya una o dos personas. Inicialmente, las mesas se cargan con valores aleatorios entre 0 y 4. Cada vez que se sientan nuevos clientes se debe mostrar el estado de las mesas. Los grupos no se pueden romper aunque haya huecos sueltos suficientes. El funcionamiento del programa se ilustra a continuación:

### Ejemplo:

Mesa nº	1	2	3	4	5	6	7	8	9	10
Ocupación	3	2	0	2	4	1	0	2	1	1

¿Cuántos son? (Introduzca -1 para salir del programa): 2

Por favor, siéntense en la mesa número 3.

Mesa nº	1	2	3	4	5	6	7	8	9	10
Ocupación	3	2	2	2	4	1	0	2	1	1

¿Cuántos son? (Introduzca -1 para salir del programa): 4

Por favor, siéntense en la mesa número 7.

Mesa nº	1	2	3	4	5	6	7	8	9	10
Ocupación	3	2	2	2	4	1	4	2	1	1

--	--	--	--	--	--	--	--	--	--	--

¿Cuántos son? (Introduzca -1 para salir del programa): 3

Tendrán que compartir mesa. Por favor, siéntense en la mesa número 6.

Mesa nº	1	2	3	4	5	6	7	8	9	10
Ocupación	3	2	2	2	4	4	4	2	1	1

¿Cuántos son? (Introduzca -1 para salir del programa): 4

Lo siento, en estos momentos no queda sitio.

Mesa nº	1	2	3	4	5	6	7	8	9	10
Ocupación	3	2	2	2	4	4	4	2	1	1

¿Cuántos son? (Introduzca -1 para salir del programa): -1

Gracias. Hasta pronto.

## ARRAYS MULTIDIMENSIONALES

### Ejercicio 16

Realiza un programa que calcule la estatura media, mínima y máxima en centímetros de personas de diferentes países. El array que contiene los nombres de los países es el siguiente: **pais = {"España", "Rusia", "Japón", "Australia"}**. Los datos sobre las estaturas se deben simular mediante un array de 4 filas por 10 columnas con números aleatorios generados al azar entre 140 y 210. Los decimales de la media se pueden despreciar. Los nombres de los países se deben mostrar utilizando el array de países (no se pueden escribir directamente)

### Ejercicio 17

Realiza un programa que sea capaz de rotar todos los elementos de una matriz cuadrada una posición en el sentido de las agujas del reloj. La matriz debe tener 12 filas por 12 columnas y debe contener números generados al azar entre 0 y 100. Se debe mostrar tanto la matriz original como la matriz resultado, ambas con los números convenientemente alineados.

### Ejercicio 18

Mejora el juego "Busca el tesoro" de tal forma que si hay una mina a una casilla de distancia, el programa avise diciendo ¡Cuidado! ¡Hay una mina cerca!

### Ejercicio 19

Realiza un programa que rellene un *array* de 6 filas por 10 columnas con números enteros positivos comprendidos entre 0 y 1000 (ambos incluidos). A continuación, el programa deberá dar la posición tanto del máximo como del mínimo.

### Ejercicio 20

Modifica el programa anterior de tal forma que no se repita ningún número en el *array*.