

Ejercicios Unidad 5: Arrays

1.- Complete las siguientes oraciones:

- a) Las listas y tablas de valores pueden guardarse en _____.
- b) El numero utilizado para referirse a un elemento específico de un array se conoce como el _____ de ese elemento.
- c) Un array que utiliza dos subíndices se conoce como un array _____.

2.- Conteste con *verdadero* o *falso* a cada una de las siguientes proposiciones; en caso de ser *falso*, explique por que.

- a) Un array puede guardar muchos tipos distintos de valores.
- b) El índice de un array debe ser generalmente de tipo float.
- c) Un elemento individual de un array que se pasa a un método y se modifica ahí mismo, contendrá el valor modificado cuando el método llamado termine su ejecución.

3.- Realice las siguientes tareas para un array llamado fracciones:

- a) Declare una constante llamada TAMANIO_ARRAY que se inicialice con 10.
- b) Declare un array con TAMANIO_ARRAY elementos de tipo double e inicialice los elementos con 0.
- c) Haga referencia al elemento 4 del array.
- d) Asigne el valor 1.667 al elemento 9 del array.
- e) Asigne el valor 3.333 al elemento 6 del array.
- f) Sume todos los elementos del array, utilizando una instrucción for. Declare la variable entera x como variable de control para el ciclo.

4.- Realice las siguientes tareas para un array llamado tabla:

- a) Declare y cree el array como un array entero con tres filas y tres columnas. Suponga que se ha declarado la constante TAMANIO_ARRAY con el valor de 3.
- b) Cuantos elementos contiene el array?
- c) Utilice una instrucción for para inicializar cada elemento del array con la suma de sus índices. Suponga que se declaran las variables enteras x e y como variables de control.

5.- Encuentre y corrija el error en cada uno de los siguientes fragmentos de programa:

- a) Suponga que `int b[] = new int[10];`
`for (int i = 0; i <= b.length; i++)`
`b[i] = 1;`
- b) Suponga que `int a[][] = { { 1, 2 }, { 3, 4 } };`
`a[1, 1] = 5;`

6.- Complete las siguientes oraciones:

- a) Un array de *m* por *n* contiene _____ filas, _____ columnas y _____ elementos.
- b) El nombre del elemento en la fila 3 y la columna 5 del array d es _____.

7.- Conteste con *verdadero* o *falso* a cada una de las siguientes proposiciones; en caso de ser *falso*, explique por que.

- a) Para referirse a una ubicación o elemento específico dentro de un array, especificamos el nombre del array y el valor del elemento específico.
- b) La declaración de un array reserva espacio para el mismo.
- c) Para indicar que deben reservarse 100 ubicaciones para el array entero p, el programador escribe la declaración

p[100];

- d) Una aplicación que inicializa con cero los elementos de un array con 15 elementos debe contener al menos una instrucción for.
- e) Una aplicación que suma el total de los elementos de un array bidimensional debe contener instrucciones for anidadas.

8.- Escriba instrucciones en Java que realicen cada una de las siguientes tareas:

- a) Mostrar el valor del elemento 6 del array f.
- b) Inicializar con 8 cada uno de los cinco elementos del array entero unidimensional g.
- c) Sumar el total de los 100 elementos del array c de punto flotante.
- d) Copiar el array a de 11 elementos en la primera porción del array b, el cual contiene 34 elementos.
- e) Determinar e imprimir los valores menor y mayor contenidos en el array w con 99 elementos de punto flotante.

9.- Considere un array entero t de dos por tres.

- a) Escriba una instrucción que declare y cree a t.
- b) ¿Cuántas filas tiene t?
- c) ¿Cuántas columnas tiene t?
- d) ¿Cuántos elementos tiene t?
- e) Escriba expresiones de acceso para todos los elementos en la fila 1 de t.
- f) Escriba expresiones de acceso para todos los elementos en la columna 2 de t.
- g) Escriba una sola instrucción que asigne cero al elemento de t en la fila 0 y la columna 1.
- h) Escriba una serie de instrucciones que inicialice cada elemento de t con cero. No utilice una instrucción de repetición.
- i) Escriba una instrucción for anidada que inicialice cada elemento de t con cero.
- j) Escriba una instrucción for anidada que reciba como entrada del usuario los valores de los elementos de t.
- k) Escriba una serie de instrucciones que determine e imprima el valor mas pequeño en t.
- l) Escriba una instrucción printf que muestre los elementos de la primera fila de t. No utilice repetición.
- m) Escriba una instrucción que totalice los elementos de la tercera columna de t. No utilice repetición.
- n) Escriba una serie de instrucciones para imprimir el contenido de t en formato tabular. Enliste los índices de columna como encabezados a lo largo de la parte superior, y enliste los índices de fila a la izquierda de cada fila.

10.- (Comisión por ventas) Utilice un array unidimensional para resolver el siguiente problema: una compañía paga a sus vendedores por comisión. Los vendedores reciben 200€ por semana mas el 9% de sus ventas totales de esa semana. Por ejemplo, un vendedor que acumule 5000€ en ventas en una semana, recibirá 200€ mas el 9% de 5000€, un total de 650€. Escriba una aplicación (utilizando un array de contadores) que, introduciéndole las ventas semanales de cada vendedor, determine cuantos vendedores recibieron salarios en cada uno de los siguientes rangos (suponga que el salario de cada vendedor se trunca a una cantidad entera):

- a) 200-299 €
- b) 300-399 €
- c) 400-499 €
- d) 500-599 €
- e) 600-699 €
- f) 700-799 €
- g) 800-899 €
- h) 900-999 €
- i) 1000 € en adelante

Sintetice los resultados en formato tabular.

11.- Escriba instrucciones que realicen las siguientes operaciones con arrays unidimensionales:

- a) Asignar cero a los 10 elementos del array cuentas de tipo entero.
- b) Sumar uno a cada uno de los 15 elementos del array bono de tipo entero.
- c) Imprimir los cinco valores del array mejoresPuntuaciones de tipo entero en formato de columnas.

12.- (Eliminación de duplicados) Use un array unidimensional para resolver el siguiente problema: escriba una aplicación que reciba como entrada cinco números, cada uno de los cuales debe estar entre 10 y 100. A medida que se lea cada numero, muéstrelo solamente si no es un duplicado de un numero que ya se haya leído. Prepárese para el “peor caso”, en el que los cinco números son diferentes. Use el array mas pequeño que sea posible para resolver este problema. Muestre el conjunto completo de valores únicos introducidos, después de que el usuario introduzca cada nuevo valor.

13.- Etiquete los elementos del array bidimensional ventas de tres por cinco, para indicar el orden en el que se establecen en cero, mediante el siguiente fragmento de programa:

```
for ( int fila = 0; fila < ventas.length; fila++ )
{
    for ( int col = 0; col < ventas[ fila ].length; col++ )
    {
        ventas[ fila ][ col ] = 0;
    }
}
```

14.- (*Tiro de dados*) Escriba una aplicación para simular el tiro de dos dados. La aplicación debe utilizar un objeto de la clase Random una vez para tirar el primer dado, y de nuevo para tirar el segundo dado. Después debe calcularse la suma de los dos valores. Cada dado puede mostrar un valor entero del 1 al 6, por lo que la suma de los valores variara del 2 al 12, siendo 7 la suma mas frecuente, mientras que 2 y 12 serán las sumas menos frecuentes. En la siguiente tabla se muestran las 36 posibles combinaciones de los dos dados.

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

Su aplicación debe tirar los dados 36,000 veces. Utilice un array unidimensional para registrar el numero de veces que aparezca cada una de las posibles sumas. Muestre los resultados en formato tabular. Determine si los totales son razonables (es decir, hay seis formas de tirar un 7, por lo que aproximadamente una sexta parte de los tiros deben ser 7).

15.- (*Sistema de reservas de una aerolínea*) Una pequeña aerolínea acaba de comprar una computadora para su nuevo sistema de reservas automatizado. Se le ha pedido a usted que desarrolle el nuevo sistema. Usted escribirá una aplicación para asignar asientos en cada vuelo del único avión de la aerolínea (capacidad: 10 asientos).

Su aplicación debe mostrar las siguientes alternativas: Por favor escriba 1 para Primera Clase y Por favor escriba 2 para Económico. Si el usuario escribe 1, su aplicación debe asignarle un asiento en la sección de primera clase (asientos 1 a 5). Si el usuario escribe 2, su aplicación debe asignarle un asiento en la sección económica (asientos 6 a 10). Su aplicación deberá entonces imprimir una tarjeta de embarque, indicando el numero de asiento de la persona y si se encuentra en la sección de primera clase o clase económica del avión.

Use un array unidimensional del tipo primitivo boolean para representar la tabla de asientos del avión. Inicialice todos los elementos del array con false para indicar que todos los asientos están vacíos. A medida que se asigne cada asiento, establezca los elementos correspondientes del array en true para indicar que ese asiento ya no esta disponible.

Su aplicación nunca deberá asignar un asiento que ya haya sido asignado. Cuando este llena la sección económica, su programa deberá preguntar a la persona si acepta ser colocada en la sección de primera clase (y viceversa). Si la persona acepta, haga la asignación de asiento apropiada. Si no acepta, imprima el mensaje "El próximo vuelo sale en 3 horas".

16.- (Ventas totales) Use un array bidimensional para resolver el siguiente problema: una compañía tiene cuatro vendedores (1 a 4) que venden cinco productos distintos (1 a 5). Una vez al día, cada vendedor pasa una nota por cada tipo de producto vendido. Cada nota contiene lo siguiente:

- a) El numero del vendedor.
- b) El numero del producto.
- c) El valor total en euros de ese producto vendido en ese día.

Así, cada vendedor pasa entre 0 y 5 notas de venta por día. Suponga que esta disponible la información sobre todas las notas del mes pasado. Escriba una aplicación que lea toda esta información para las ventas del ultimo mes y que resuma las ventas totales por vendedor, por producto. Todos los totales deben guardarse en el array bidimensional ventas.

Después de procesar toda la información del mes pasado, muestre los resultados en formato tabular, en donde cada columna represente a un vendedor especifico y cada fila represente a un producto. Saque el total de cada fila para obtener las ventas totales de cada producto durante el ultimo mes. Saque el total de cada columna para obtener las ventas totales de cada vendedor durante el ultimo mes. Su impresión tabular debe incluir estos totales cruzados a la derecha de las filas totalizadas, y en la parte inferior de las columnas totalizadas.

17.- (*Gráficos de tortuga*) El lenguaje Logo hizo famoso el concepto de los *gráficos de tortuga*. Imagine a una tortuga mecánica que camina por todo el cuarto, bajo el control de una aplicación en Java. La tortuga sostiene una pluma en una de dos posiciones, arriba o abajo. Mientras la pluma esta abajo, la tortuga va trazando figuras a medida que se va moviendo, y mientras la pluma esta arriba, la tortuga se mueve alrededor libremente, sin trazar nada. En este problema usted simulara la operación de la tortuga y creara un bloc de dibujo computarizado.

Utilice un array de 20 por 20 llamado piso, que se inicialice con ceros. Lea los comandos que introduzca el usuario. Lleve el registro de la posición actual de la tortuga en todo momento, y si la pluma se encuentra arriba o abajo. Suponga que la tortuga siempre empieza en la posición (0, 0) del piso, con su pluma hacia arriba. El conjunto de comandos de la tortuga que su aplicación debe procesar se muestra en la siguiente figura:

Comando Significado	
1	Pluma arriba
2	Pluma abajo
3	Girar a la derecha
4	Girar a la izquierda
5	Avanzar hacia delante 10 espacios
10	(cambiar 10 por el número de espacios deseados)
6	Imprimir el array de 20 por 20
9	Fin

Suponga que la tortuga se encuentra en algún lado cerca del centro del piso. El siguiente “programa” dibuja e imprime un cuadrado de 12 por 12, dejando la pluma en posición levantada:

```

2
5
12
4
5
12
4
5
12
4
5
12
1
6
9

```

A medida que la tortuga se vaya desplazando con la pluma hacia abajo, asigne 1 a los elementos apropiados del array piso. Cuando se de el comando 6 (imprimir el array), siempre que haya un 1 en el array muestre un asterisco o cualquier carácter que usted elija. Siempre que haya un 0, muestre un carácter en blanco.

18.- (Ocho reinas) Un enigma para los entusiastas del ajedrez es el problema de las Ocho reinas, el cual pregunta lo siguiente: ¿es posible colocar ocho reinas en un tablero de ajedrez vacío, de tal manera que ninguna “ataque” a cualquier otra (es decir, que no haya dos reinas en la misma fila, en la misma columna o a lo largo de la misma diagonal)? Diseñe una aplicación que, probando todas las posibilidades, determine si existe alguna combinación de posiciones en la que las ocho reinas puedan estar en el tablero sin atacarse.

19.- (La criba de Eratóstenes) Un número primo es cualquier entero mayor que 1, divisible solo por sí mismo y por el número 1. La Criba de Eratóstenes es un método para encontrar números primos, el cual opera de la siguiente manera:

- a) Cree un array del tipo primitivo boolean, con todos los elementos inicializados en true. Los elementos del array con índices primos permanecerán como true. Cualquier otro elemento del array eventualmente cambiara a false.
- b) Empezando con el índice 2 del array, determine si un elemento dado es true. De ser así, itere a través del resto del array y asigne false a todo elemento cuyo índice sea múltiplo del índice del elemento que tiene el valor true. Después continúe el proceso con el siguiente elemento que tenga el valor true. Para el índice 2 del array, todos los elementos más allá del elemento 2 en el array que tengan índices múltiplos de 2 (los índices 4, 6, 8, 10, etcétera) se establecerán en false; para el índice 3 del array, todos los elementos más allá del elemento 3 en el array que tengan índices múltiplos de 3 (los índices 6, 9, 12, 15, etcétera) se establecerán en false; y así sucesivamente.

Cuando este proceso termine, los elementos del array que aun sean true indicaran que el índice es un número primo. Estos índices pueden mostrarse. Escriba una aplicación que utilice un array de 1000 elementos para determinar e imprimir los números primos entre 2 y 999. Ignore los elementos 0 y 1 del array.

20.- (Simulación: la tortuga y la liebre) En este problema usted recreara la clásica carrera de la tortuga y la liebre. Utilizara la generación de números aleatorios para desarrollar una simulación de este memorable suceso.

Nuestros competidores empezaran la carrera en la posición 1 de 70 posiciones. Cada posición representa a una posible posición a lo largo del curso de la carrera. La línea de meta se encuentra en la posición 70. El primer competidor en llegar a la posición 70 recibirá una cubeta llena con zanahorias y lechuga frescas. El recorrido se abre paso hasta la cima de una resbalosa montaña, por lo que ocasionalmente los competidores pierden terreno.

Un reloj hace tictac una vez por segundo. Con cada tic del reloj, su aplicación debe ajustar la posición de los animales de acuerdo con las reglas de la siguiente tabla. Use variables para llevar el registro de las posiciones de los animales (los números son del 1 al 70). Empiece con cada animal en la posición 1 (la “puerta de inicio”). Si un animal se resbala hacia la izquierda antes de la posición 1, regréselo a la posición 1.

Genere los porcentajes en la figura de movimientos siguiente produciendo un entero aleatorio i en el rango $1 \leq i \leq 10$. Para la tortuga, realice un “paso pesado rápido” cuando $1 \leq i \leq 5$, un “resbalón” cuando $6 \leq i \leq 7$ o un “paso pesado lento” cuando $8 \leq i \leq 10$. Utilice una técnica similar para mover a la liebre.

Animal	Tipo de movimiento	Porcentaje del tiempo	Movimiento actual
Tortuga	Paso pesado rápido	50 %	3 posiciones a la derecha
	Resbalón	20 %	6 posiciones a la izquierda
	Paso pesado lento	30 %	1 posición a la derecha
Liebre	Dormir	20 %	Ningún movimiento
	Gran salto	20 %	9 posiciones a la derecha
	Gran resbalón	10 %	12 posiciones a la izquierda
	Pequeño salto	30 %	1 posición a la derecha
	Pequeño resbalón	20 %	2 posiciones a la izquierda

Empiece la carrera imprimiendo el mensaje

PUM!!!

Y ARRANCAN!!!

Luego, para cada tic del reloj (es decir, cada repetición de un ciclo) imprima una línea de 70 posiciones, mostrando la letra T en la posición de la tortuga y la letra H en la posición de la liebre. En ocasiones los competidores se encontraran en la misma posición. En este caso, la tortuga muerde a la liebre y su aplicación debe imprimir OUCH!!! empezando en esa posición. Todas las posiciones de impresión distintas de la T, la H o el mensaje OUCH!!! (en caso de un empate) deben estar en blanco.

Después de imprimir cada línea, compruebe si uno de los animales ha llegado o se ha pasado de la posición 70. De ser así, imprima quien fue el ganador y termine la simulación. Si la tortuga gana, imprima LA TORTUGA GANA!!! YAY!!! Si la liebre gana, imprima La liebre gana. Que mal. Si ambos animales ganan en el mismo tic del reloj, tal vez usted quiera favorecer a la tortuga (la mas débil) o tal vez quiera imprimir Es un empate. Si ninguno de los dos animales gana, ejecute el ciclo de nuevo para simular el siguiente tic del reloj. Cuando este listo para ejecutar su aplicación, reúna a un grupo de aficionados para que vean la carrera. !Se sorprenderá al ver lo participativa que puede ser su audiencia!

21.- Escriba una aplicación que utilice la generación de números aleatorios para crear enunciados. Use cuatro arrays de cadenas llamados articulo, sustantivo, verbo y preposición. Cree una oración seleccionando una palabra al azar de cada uno de los arrays, en el siguiente orden: articulo, sustantivo, verbo, preposición., articulo y sustantivo. A medida que se elija cada palabra, concaténela con las palabras anteriores en el enunciado. Las palabras deberán separarse mediante espacios. Cuando se muestre el enunciado final, deberá empezar con una letra mayúscula y terminar con un punto. El programa deberá generar 20 enunciados y mostrarlos en un área de texto.

El array de artículos debe contener los artículos "el", "un", "algún" y "ningún"; el array de sustantivos deberá contener los sustantivos "niño", "niña", "perro", "ciudad" y "coche"; el array de verbos deberá contener los verbos "condujo", "salto", "corrió", "camino" y "omitió"; el array de preposiciones deberá contener las preposiciones "a", "desde", "encima de", "debajo de" y "sobre".