



TEMA 6: Aplicación de las estructuras de almacenamiento

PROGRAMACIÓN

Beatriz López Palacio

EJERCICIOS 6.8.

1. Crear una matriz de 3x3 con los números del 1 al 9. Mostrarla por pantalla, tal como se representa en una matriz.
2. Crear una matriz de 5 filas y n columnas (se solicitará la dimensión al usuario). Rellenar dicha matriz con números aleatorios entre 0 y 10.
3. Crear dos matrices de nxm y sumar sus valores, los resultados se deben almacenar en otra matriz. Los valores y la dimensión, serán insertados por el usuario. Mostrar las matrices originales y el resultado.
4. Crear una matriz 4x4 de números enteros que inicialmente está vacía, nos piden hacer un menú con estas opciones:
 1. *Rellenar*: Rellenar toda la matriz de números, deben solicitarse al usuario por teclado.
 2. *SumaFila*: Suma de los datos de una fila que el usuario elegirá (controlar que elija una correcta).
 3. *SumaColumna*: Suma de los datos de una columna que el usuario elegirá (controlar que elija una correcta).
 4. *SumaPrincipal*: Sumar la diagonal principal.
 5. *SumaInversa*: Sumar la diagonal inversa.
 6. *Media*: Calcular la media de todos los valores de la matriz.

Tener en cuenta que hasta que no se realice la primera opción, el resto de opciones no se deberá ejecutar, simplemente mostrar un mensaje donde diga que debes rellenar la matriz.

5. Crea dos arrays multidimensionales de 2x3 y rellénalos como quieras (números aleatorios o por teclado).

Haz un método que sume las matrices, es decir la posición 0,0 de la matriz 1 con la posición 0,0 de la matriz 2 y así sucesivamente, este método no debe devolver nada.

Realiza un método para imprimir, al que llamar 3 veces, para ir mostrando por pantalla el contenido de cada matriz, así como la matriz suma resultante.

6. Se desea realizar una encuesta a 10 personas, en esta encuesta indicaremos el sexo (1=masculino, 2=femenino), si trabaja (1=si trabaja, 2=no trabaja) y su sueldo (si tiene un trabajo, sino será un cero) estará entre 600 y 2000 (valor entero). Los valores pueden ser generados aleatoriamente. Calcula y muestra lo siguiente:
 1. Porcentaje de hombres (tengan o no trabajo).
 2. Porcentaje de mujeres (tengan o no trabajo).
 3. Porcentaje de hombres que trabajan.
 4. Porcentaje de mujeres que trabajan.
 5. El sueldo promedio de los hombres que trabajan.
 6. El sueldo promedio de las mujeres que trabajan.
7. Escribe una interfaz, llamada ColeccionCoches, que implemente los siguientes métodos:
 1. *EstaVacia*: Devuelve verdadero si la colección de coches está vacía y falso en caso contrario.
 2. *Existe*: Devuelve verdadero si el vehículo solicitado por teclado pertenece a la colección.
 3. *Añadir*: Añade a la colección el coche solicitado por teclado.
 4. *Mostrar*: Muestra los objetos que tiene en ese momento la colección.
8. Implementar un método capaz de obtener (seleccionando la posición del elemento del array a buscar de forma aleatoria) un elemento de un array del tipo que sea.

En el programa principal se inicializará el array, que se pasarán a dicho método para que este pueda llevar a cabo su elección y posteriormente y también en el main imprimirlo en pantalla.
9. Crea una clase colegio que almacene la nacionalidad de sus alumnos. La clase realizará las siguientes opciones:
 1. *Añadir*: Añadir la nacionalidad de un nuevo alumno.
 2. *Mostrar*: Mostrar las distintas nacionalidades y el número de alumnos.
 3. *Seleccionar*: Dejar únicamente la nacionalidad que el usuario decida por teclado.

10. Crear una colección semana que contenga los siete días de la semana. A continuación copiar ésta a otra colección llamada horario:

- Mostrar el contenido de la lista original.
- Elimina el sábado y el domingo del horario.
- Muestra el contenido actual del horario.
- Busca si existe Monday en el horario e indícalo con un mensaje.
- Mantén en semana sólo los días del horario.

11. Crea un conjunto al que se va a llamar jugadores. Inserta en el conjunto los jugadores del Sporting: Manu, Carmona, Murilo, Aitor, Nacho.

- Realiza un bucle y muestra sus nombres.
- Consulta si en el conjunto existe el jugador que desee el usuario. Avisa si existe o no.
- Crea un segundo conjunto jugadores2 con los jugadores Manu y Murilo.
- Consulta si todos los elementos de jugadores2 existen en jugadores.
- Realiza una unión de ambos conjuntos en jugadores.
- Elimina los jugadores del conjunto jugadores2 y comprueba el número de elementos que tiene ahora.

12. Crea un programa que almacene la nota final de 10 alumnos de una colección, dónde el alumno se identificará por su nombre.