



## Laboratorio de Estructuras de Datos

### Práctica 3. Arreglos unidimensionales numéricos

Unidad Temática: 1. Introducción a las estructuras de datos y estructuras fundamentales

📖 👤 Profesor: Dr. Aldonso Becerra Sánchez

#### Índice

1	Objetivo de la tarea	1
2	Tiempo aproximado de realización	1
3	Fecha de entrega	1
4	Fecha de entrega con extensión y penalización	1
5	Introducción	1
6	Actividades a realizar	1
6.1	Actividad Inicial	1
6.2	Actividad 1	1
6.3	Actividad 2	1
6.4	Actividad 3	2
6.5	Actividad 4	2
6.6	Actividad 5	2
6.7	Actividad 6	2
6.8	Actividad 7	2
7	Contáctame	2
	References	2

#### 6. Actividades a realizar

##### 6.1. Actividad Inicial

Lea primero toda la práctica ⚠️. No inicie a programar sin leer todo cuidadosamente primero. Recuerde que debe generar el reporte en formato IDC con todos sus componentes.

##### 6.2. Actividad 1

Primero genere la Introducción 📝.

##### 6.3. Actividad 2

#### Información importante

Esta actividad debe entrar en la parte de *Desarrollo* 📄.

Complete el TDA llamado “ArregloListaInfoEstaticaNumeros”. Esta clase tiene como objetivo manejar puros valores numéricos, es decir que hereden de la clase Number (utiliza el operador instanceof para facilitar este proceso) cada valor que contendrá. Defina esta clase para que herede de ArregloListaInfoEstatica.

Se le pide entonces:

- Para ello deberá validar que el contenido que se intenta agregar, eliminar, buscar, modificar y demás operaciones heredadas de ArregloListaInfoEstatica sean de la clase Number. Para hacer eso tiene que sobre-escribir los métodos necesarios y hacer las invocaciones necesarias a la super clase. Minimice el código reescrito. Si cualquier valor proporcionado no es numérico, debe prohibir la acción del método. Se recomienda convertir todo valor numérico a double, ya que es el máximo en tamaño.
- boolean porEscalar(Number escalar):** multiplicar el escalar dado por cada posición del arreglo numérico. Regrese falso si el arreglo actual está vacío.
- boolean sumarEscalar(Number escalar):** sumar el escalar dado a cada posición del arreglo numérico. Regrese falso si el arreglo actual está vacío.
- boolean sumar(ArregloListaInfoEstaticaNumeros arreglo2):** sumar la posición 1 del arreglo actual con la posición 1 de arreglo2, y así sucesivamente. Note que los valores pueden ser negativos, en tal caso debe seguir funcionando este proceso. Valide dimensiones.
- boolean multiplicar(ArregloListaInfoEstaticaNumeros arreglo2):** que haga producto de posición 1 del arreglo actual por posición 1 de arreglo2, y así sucesivamente. Regrese falso si el arreglo actual está vacío.

#### 1. Objetivo de la tarea

Creación de TDA arreglo (ArregloListaInfoEstaticaNumeros) para su posterior uso en aplicaciones comunes.

#### 2. Tiempo aproximado de realización

🕒 5 horas.

#### 3. Fecha de entrega

📅 20 agosto de 2024.

#### 4. Fecha de entrega con extensión y penalización

📅 21 agosto de 2024, 7:30 am.

#### 5. Introducción


Existe diversidad de usos que se les puede dar a los arreglos que guardan solo números. Desde este punto de vista, los arreglos propician que muchos planteamientos puedan tener una solución sencilla si se llevan a cabo con la ayuda de ellos. [1], [2], [3].

52 ■ **boolean aplicarPotencia(Number escalar):** que haga la operación de potencia de cada elemento del arreglo (base) por el  
 53 exponente pasado como escalar. Regrese falso si el arreglo actual  
 54 está vacío.  
 55  
 56 ■ **boolean aplicarPotencia(ArregloListaInfoEstaticaNumeros  
 57 arregloEscalares):** que haga la operación de potencia de cada  
 58 elemento del arreglo (base) por el exponente pasado como  
 59 arreglo, posición por posición. Valide dimensiones.  
 60 ■ **double productoEscalar(ArregloListaInfoEstaticaNumeros  
 61 arreglo2):** por ejemplo considere los vectores matemáticos  
 62 (arreglos) siguientes de dimensión 3:  $a, b, c \cdot x, y, z$   $T = a \cdot x$   
 63  $+ b \cdot y + c \cdot z$ . Donde la T es la transpuesta del vector y  $\cdot$  es la  
 64 multiplicación elemento por elemento.  
 65 ■ **double normal2():** saca la magnitud / módulo / norma L2  
 66 del vector (arreglo); por ejemplo considere el siguiente vector  
 67 numérico de dimensión 3:  $V = a, b, c, ||V|| = \sqrt{a^2 + b^2 + c^2}$ .  
 68 Donde sqrt es la raíz cuadrada.  
 69 ■ **double normaEuclidiana(ArregloListaInfoEstaticaNumeros  
 70 arreglo2):** debe calcular la norma euclidiana del vector nu-  
 71 mérico AB (arreglos n dimensionales). Este cálculo está dado  
 72 por:  
 73 donde  $A = a_1, a_2, a_3, \dots, a_n$  y  $B = b_1, b_2, b_3, \dots, b_n$ .  
 74 ■ **double sumarArregloEstatico(ArregloListaInfoEstatica  
 75 listasArreglos):** debe sumar de uno por uno un conjunto de  
 76 arreglos de tipo ArregloListaInfoEstaticaNumeros almacenados  
 77 en la variable listasArreglos al arreglo actual. Cada posición de  
 78 listas guarda un arreglo.  
 79 ■ **double sumarEscalares(ArregloListaInfoEstaticaNumeros  
 80 escalares):** debe sumar de uno por uno un conjunto de escalares  
 81 almacenados en la variable escalares al arreglo actual. Escalares  
 82 es un arreglo que guarda en cada posición un escalar.  
 83 ■ **double sumarIndices(ArregloListaInfoEstaticaNumeros  
 84 arregloIndice):** debe sumar, del arreglo actual, las posiciones  
 85 de él que indica el arreglo llamado arregloIndice, el cual alma-  
 86 cena las posiciones que se deben tomar del arreglo actual para  
 87 hacer la suma.  
 88 ■ **ArregloListaInfoEstatica subArre-  
 89 glo(ArregloListaInfoEstaticaNumeros arregloIndice):**  
 90 debe regresar un arreglo conteniendo los elementos del arreglo  
 91 actual que se obtienen del arreglo de índices "arregloIndice",  
 92 el cual contiene las posiciones de los índices de donde se  
 93 obtendrán los datos a retornar. Significa que se deberán hacer  
 94 dos métodos, el correspondiente en la super clase (ArregloLis-  
 95 taInfoEstatica), para datos generales y el correspondiente a la  
 96 clase ArregloListaInfoEstaticaNumeros, correspondiente solo a  
 97 números.  
 98 ■ Dos vectores son ortogonales si y sólo si su producto escalar es  
 99 nulo. Es decir:  $u \cdot v = 0$ . Genere el método **boolean esOrtogo-  
 100 nal(ArregloListaInfoEstaticaNumeros arreglo2)** para deter-  
 101 minar si el arreglo actual es ortogonal al arreglo pasado como  
 102 argumento.  
 103 ■ Dos vectores son paralelos si y sólo si uno de ellos es múltiplo  
 104 escalar del otro. Es decir:  $u = kv$ . Genere el método **boolean  
 105 esParalelo(ArregloListaInfoEstaticaNumeros arreglo2)** pa-  
 106 ra determinar si el arreglo actual es paralelo al arreglo pasado  
 107 como argumento.


## 6.4. Actividad 3

109 Pruebe el funcionamiento del programa de la actividad 2 con todo y  
 110 sus capturas de pantalla.


## 6.5. Actividad 4

112 Realice la sección de Código agregado  (diagrama de clases UML).





## 6.6. Actividad 5

Realice la sección de Pre-evaluación  (use los lineamientos esta-  
 114 blecidos). 115

## 6.7. Actividad 6

Finalmente haga las Conclusiones .


## 6.8. Actividad 7


Subir los entregables (pdf  y zip  con código )  a Moodle. 119

## 7. Contáctame

Puedes contactarme a través de los siguientes medios.

 <https://moodle.ingsoftware.uaz.edu.mx/>

 a7donso@gmail.com

 Cubículo

 Salón CC2-IS

## Referencias

[1] O. Cairo y S. Guardati, *Estructura de datos*. McGraw-Hill.

[2] L. Joyanes Aguilar, *Fundamentos de programación, algoritmos  
 u estructura de datos*. McGraw-Hill.

[3] M. A. Weiss, *Estructura de datos en Java*. Addison Wesley.