

# Estructuras de Datos

## Lección 1 - Paralelo 02

### PAO II 2021

## Instrucciones

En esta lección, **usted** es responsable de definir el prototipo de cada método solicitado en este documento. Es decir, usted debe decidir cuántos argumentos se requieren (y el tipo de cada uno) así como el tipo de dato que el método debe retornar.

Cuando considere pertinente, incluya comentarios en sus respuestas para proveer detalles adicionales sobre sus decisiones de diseño (p. ej., *// esta implementación asume que...*).

En esta lección **NO** se solicita probar sus implementaciones en un programa principal.

Sus respuestas deben ser escritas **directamente en el AulaVirtual**. **NO** está permitido el uso de NetBeans o cualquier otro IDE.

Las respuestas se recibirán hasta las 12:05. No se admitirán respuestas enviadas a destiempo. Para evitar inconvenientes, planifique el envío de sus respuestas considerando la calidad de su conexión a internet.

**Los métodos solicitados se detallan en la siguiente página**

- 1) Implemente el método `insertarOrdenado`, de la clase `LinkedList`, que recibe un objeto y lo inserta en una lista de manera ordenada ascendentemente.

**Ejemplo 1:**

Si una lista (que originalmente está vacía) invoca el método `insertarOrdenado` con los números 7, 2, 5, 0 y 6, la lista resultante es `[0, 2, 5, 6, 7]`

**Ejemplo 2:**

Si una lista (que originalmente está vacía) invoca el método `insertarOrdenado` con las personas `{Raquel, 40}`, `{Isabel, 48}` y `{Félix, 42}`, debería ser posible generar cualquier de las siguientes listas:

```
// inserción ordenada por alfabéticamente, de acuerdo al nombre:  
[{"Félix", 42}, {"Isabel", 48}, {"Raquel", 40}]  
  
// inserción ordenada por edad:  
[{"Raquel", 40}, {"Félix", 42}, {"Isabel", 48}]
```

- 2) Implemente el método `removeElementos`, de la clase `ArrayList`, que recibe una lista de enteros. El método debe remover de la lista que lo invoca a los elementos en las posiciones indicadas en la lista que se recibe como argumento.

**Ejemplo 1:**

Si la lista `["One", "Two", "Three", "Four", "Five"]` invoca a `removeElementos` con el argumento `[3, 1, 4]` ésta queda como `["One", "Three"]`

**Ejemplo 2:**

Si la lista `[10, 3, 25, 4, 8]` invoca a `removeElementos` con el argumento `[0, 2]`, ésta queda como `[3, 4, 8]`

- 3) Implemente el método estático `sumarNumerosGrandes`, que recibe dos listas de enteros. Cada una de las listas recibidas representa un número *grande*. Por ejemplo, la lista `[1, 8, 7, 6, 7, 3, 3]` representa el número 1876733. El método `sumarNumerosGrandes` debe retornar una nueva lista que representa la suma de las dos listas recibidas como argumentos.

**Ejemplo 1:**

El método `sumarNumerosGrandes` invocado con las listas `[7, 5, 9, 4, 8]` y `[8, 7]` retorna la lista `[7, 6, 0, 3, 5]`

**Ejemplo 2:**

El método `sumarNumerosGrandes` invocado con las listas `[3, 1, 4]` y `[1, 1, 1]` retorna la lista `[4, 2, 5]`