

ESTRUCTURAS DE DATOS

Curso 2025-26

PRÁCTICA 2

Uso de iteradores

Instrucciones

- Se debe completar en una sesión.
- Práctica individual.
- Lee el enunciado completo antes de comenzar. Los comentarios incluidos en el código también proporcionan información útil y necesaria.
- Al finalizar la práctica debes entregar el código desarrollado.
- La práctica será APTA si se superan todos los test de validación proporcionados.

En esta práctica se va a trabajar en el uso de iteradores para acceder y manipular los datos contenidos en distintos tipos de colecciones. Se plantean ejercicios para usar iteradores básicos, definidos en la clase `Iterator<T>`, e iteradores de listas, `ListIterator<T>`.

Se te proporciona el fichero `Iteradores.java`, y los ejercicios piden que implementes varios métodos estáticos de la clase `Iteradores`. También se te proporciona el fichero `IteradoresJUnit.java` que implementa los test de JUnit4 que comprueban el correcto funcionamiento de los métodos implementados.

Ejercicio 1

Implementa un método que tome dos conjuntos de enteros, y los modifique de tal manera que al terminar el primer conjunto contenga los elementos pares de ambos, y el segundo los impares:

```
public static void partir(Set<Integer> pares, Set<Integer> impares)
```

Por ejemplo:

pares: (5, 2, 9, 10, 13)		pares: (2, 10, 4, 6, 0)
impares: (4, 6, 5, 0, -1)		impares: (5, 9, 13, -1)

Ejercicio 2

Escribe un método que intercambie las posiciones pares e impares de una lista de enteros. El método tomará como parámetro un iterador de listas, `ListIterator`.

```
public static int intercambio(ListIterator<Integer> iter)
```

El iterador podrá estar inicialmente en cualquier posición de la lista. El método devuelve como resultado el número de elementos de la lista. Por ejemplo:

lista: [9, 3, 4, 0, 1, 2, 2, 8]		lista: [3, 9, 0, 4, 2, 1, 8, 2] Resultado: 10
---------------------------------	---	---

Ejercicio 3

Escribe un método que tome una lista de enteros y traslade elementos que ocupen una posición múltiplo de tres al final de la misma lista, preservando el orden en el que aparecen dentro de la ella. El método tomará como parámetro un iterador de listas, `ListIterator`.

```
public static int trespatras(ListIterator<Integer> iter)
```

El iterador podrá estar inicialmente en cualquier posición de la lista. El primer elemento de la lista ocupa la posición 1. El método devuelve como resultado el número de elementos en posición múltiplo de tres trasladados al final de la lista. Por ejemplo:

```
lista: [9, 3, 4, 0, 1, 2, 2, 8 ]      ➡      lista: [9, 3, 0, 1, 2, 8, 4, 2]      Resultado: 2
```

Ejercicio 4

Escribe un método que tome como parámetro un iterador de listas y agrupe los elementos repetidos que se encuentran en la lista.

```
public static<T> void agruparRepetidos(ListIterator<T> iter)
```

El iterador podrá estar inicialmente en cualquier posición de la lista. Por ejemplo:

```
lista: [5, 6, 3, 5, 3, 9], iter en pos 4 ➡ lista: [5, 5, 6, 3, 3, 9]
```

```
lista: [9, 1, 3, 5, 3, 1, 3, 9], iter en pos 2 ➡ lista: [9, 9, 1, 1, 3, 3, 3, 5]
```

```
lista: [5, 6, 3], iter en pos 0 ➡ lista: [5, 6, 3]
```