

Facultad de Ciencias, UNAM

Práctica 2: Lista

Alejandro Hernández Mora

Jerónimo Almeida Rodríguez

Adriana Sánchez del Moral

Kethrim Guadalupe Trad Mateos

Fecha de entrega: jueves 9 de marzo de 2023

Descripción

La entrega de esta práctica es individual.

Lee con atención las instrucciones y resuelve.

En esta práctica, trabajarán con una lista que implementa a la interface `Listable` con los siguientes métodos:

- `public int indiceDe(T elemento);`
- `public T getElemento(int i) throws IndexOutOfBoundsException`
- `public Lista<T>reversa()`
- `public Lista<T>copiar()`

A su vez, `Listable` implementa la interface `Coleccionable` con los siguientes métodos:

- `public void agregar(T elemento) throws IllegalArgumentException;`
- `public void eliminar(T elemento) throws NoSuchElementException;`
- `public boolean contiene(T elemento);`
- `public boolean esVacia();`
- `public void vaciar();`
- `public int getTamano();`
- `public java.util.Iterator iterator();`

A diferencia del conjunto, cuya estructura subyacente era un arreglo sobre el cual trabajabamos y hacíamos todas las operaciones, la estructura subyacente de una lista es un objeto nodo que contiene a un elemento y dos referencias, una al nodo siguiente y otra al nodo anterior.

Antes de iniciar la práctica, hay que ejecutar los comandos:

- `$ javadoc -d docs Coleccionable.java`
- `$ javadoc -d docs Listable.java`
- `$ javadoc -d docs Lista.java`

Y abrir en su navegador el archivo `allclasses-index.html`, donde encontrarán la documentación de las interfaces `Coleccionable` y `Listable` y de la clase `Lista`. Una vez hayan leído la documentación, ver qué es lo que se pide que se haga con cada método. Su práctica consiste en implementar los métodos de la clase `Lista.java` según la descripción que les es dada en la documentación.

Condiciones de entrega

1. Descargar el repositorio que les fue asignado desde la siguiente liga: <https://classroom.github.com/a/9vfGV17N>
2. Programar los métodos especificados en la clase Lista.java y verificar que estos funcionen adecuadamente.
3. Mantener una estructura limpia de su práctica, con el código en la carpeta **src** y los documentos en la carpeta **docs**
4. Su código debe seguir buenas prácticas de programación, ser legible y estar bien indentado.
5. Subir su código a github.
6. Subir a github un archivo readme.md con su nombre completo, descripción del trabajo y pruebas de ejecución.
7. Entregar tarea en google Classroom (no es necesario subir nada)