

UNIVERSIDAD NACIONAL AUTÓNOMA DE
MÉXICO
Facultad de Ciencias



Estructuras de Datos

Práctica 5: Lista ligada parte I

Profesora:

Amparo López Gaona

Ayudante: Adrián Aguilera Moreno

Ayudante de Laboratorio: Kevin Jair Torres Valencia

Objetivo

El objetivo de esta práctica es que los estudiantes comprendan y desarrollen la estructura de datos de una Lista Simplemente Ligada en Java, comenzando con la implementación de una interfaz genérica llamada "Coleccion". Esta interfaz definirá los métodos fundamentales que cualquier colección debe implementar.

Introducción

Las estructuras de datos son fundamentales en la programación, ya que permiten organizar y manipular la información de manera eficiente. Una de las estructuras más utilizadas es la lista simplemente ligada, la cual ofrece una forma dinámica de almacenar elementos sin la necesidad de definir un tamaño fijo desde el inicio.

Una lista ligada es una agrupación de objetos denominados nodos, no necesariamente almacenados en forma adyacente. Cada nodo contiene el elemento (o dato) y un enlace o liga a su sucesor. El último elemento tendrá un enlace con valor null para indicar que no tiene sucesor.

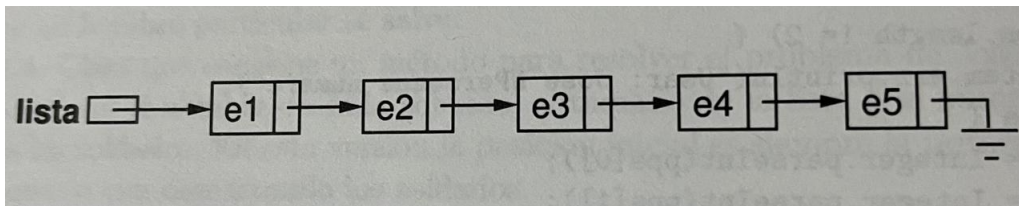


Figure 1: Lista ligada con cinco nodos.

Una lista ligada suele tener un principio y un fin, pero no se puede llegar en tiempo constante a cualquier elemento por su posición. El acceso a los elementos en la lista es secuencial, es decir, para acceder a cualquier elemento es necesario recorrer secuencialmente los elementos en la lista hasta llegar al elemento deseado.

Las listas ligadas, como se han descrito, pueden variar al implementarse, si se decide tener un apuntador al inicio de la lista, un apuntador al final de la lista, si la lista terminará con el valor null, como se mostro en la figura, o bien con un nodo especial, si se usará una sola liga o bien dos.

Desarrollo

En esta práctica, comenzarán con la creación de una interfaz genérica llamada "Coleccion" que extiende a `Iterable<T>` que definirá los métodos esenciales de manipulación de una colección de elementos. Posteriormente, desarrollarán la estructura de la lista simplemente ligada, asegurándose de comprender los conceptos de nodos.

Para la implementación de la interfaz, para cada operación se espera el siguiente comportamiento:

- **insertar:** Agrega a la lista el elemento especificado como parámetro, sin importar la posición en que lo haga y sin importar si ya existe un elemento igual a él en la lista. El tamaño de la lista crece en una unidad.
- **eliminar:** Si el elemento existe en la lista lo elimina, en caso contrario no hace nada. Si la operación es exitosa se reduce el tamaño de la lista en una unidad.
- **acceder:** Devuelve el elemento en la posición indicada.
- **buscar:** Busca un elemento en la colección, regresa `true` si el elemento está en la lista y `false` en otro caso. Este método no cambia el estado de la lista.
- **vaciar:** Elimina todos los elementos de la lista. El tamaño de la lista después de esta operación es cero.

Nota: Para la entrega de esta práctica solo basta con la Interfaz anterior, no se tiene que realizar la implementación de la lista por el momento.

Formato de Entrega

1. Las prácticas se entregarán en parejas.
2. NO incluir los archivos .class dentro de la carpeta.
3. Los archivos de código fuente deben estar documentados.
4. Se pueden discutir y resolver dudas entre los integrantes del grupo. Pero cualquier práctica plagiada total o parcialmente será penalizada con cero para los involucrados.
5. La práctica se debe subir al Github Classroom correspondiente.
6. La entrega en classroom debe contener el link HTTPS y SSH de su repositorio y es lo único que se debe entregar.
7. El horario y día de entrega se acordará en la clase de laboratorio y no deberá sobrepasar 2 clases de laboratorio.