Estructura de Datos y Algoritmos 1

Tema 5 - Estructuras de datos NO explícitas

ArrayList <T>

LinkedList <T>

Buscar indexOf (T aux)
↳ por

ED ASOCIADAS

TreeSet <T> — HashSet <T>
↳ NO TIENE ITERADOR FUERA

TreeMap <T> — HashMap <K, V>

Búsqueda → Clave

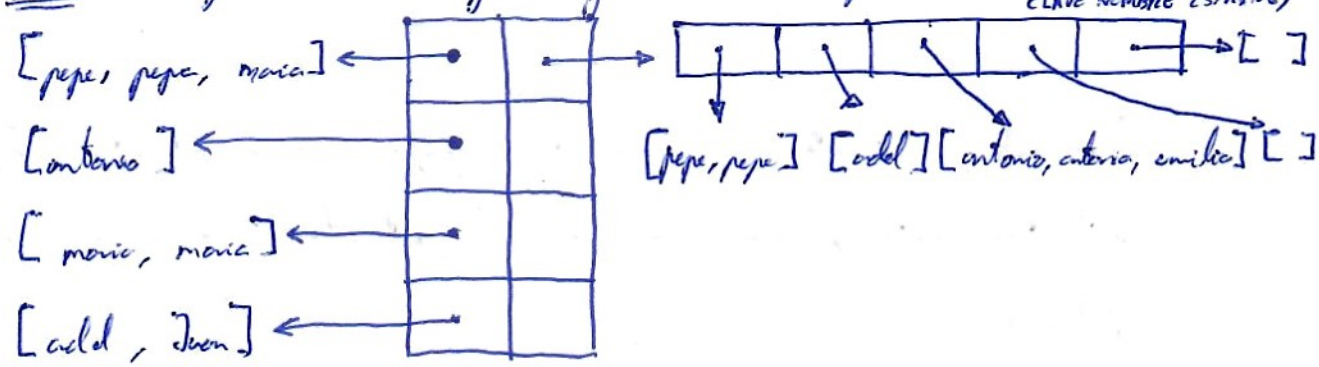
Búsqueda → referencia a valor → getKey (Clave)

TreeMap <Estudiante, estudiantes>
↓ aux — ↓ Objeto Completo

Cosas de java

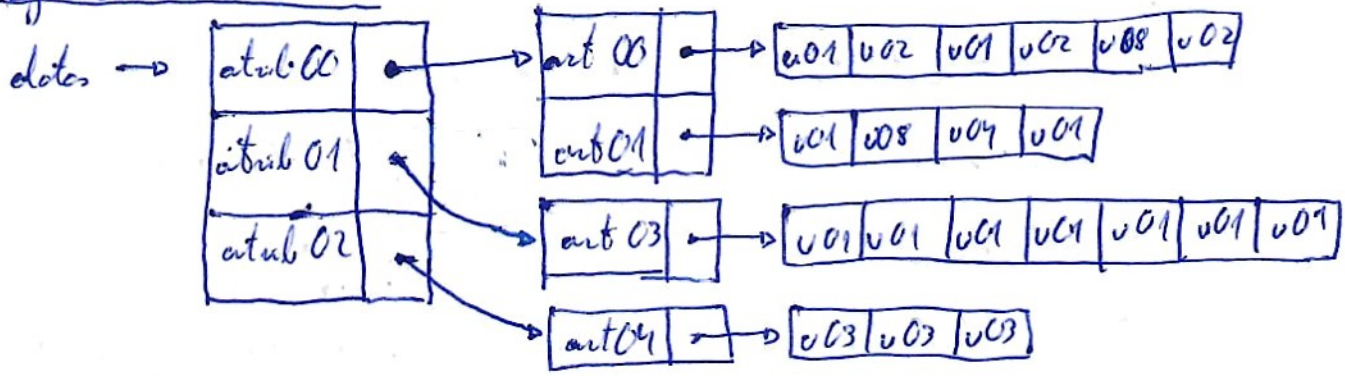
HashMap <Nodo, HashSet <Nodo>> → Grafo NO dirigido (N : M)

Nodo encapsula un ArrayList genérico de Objeto T T ≡ Persona
CLAVE NOMBRE (STRING)

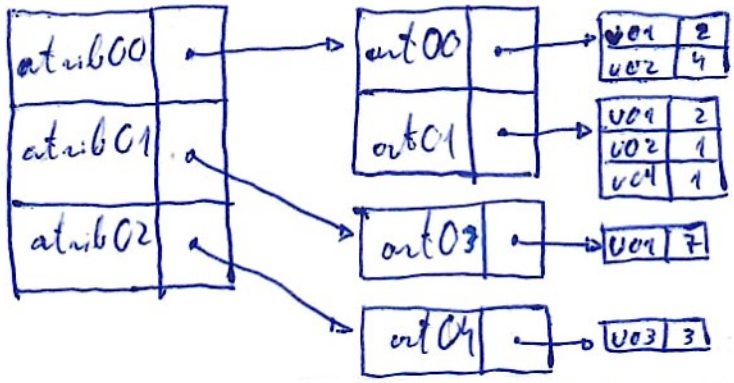


Dominio de aplicación → Red Social por ejemplo.

Ejercicio de examen



Se genera convertir en: Convertir()



TreeMap<K1 String, TreeMap<K2 String, ArrayList<K3 Integer>>>

TreeMap<K4 String, TreeMap<K5 String, TreeMap<K6 String, Integer>>>
V6

Teniendo en cuenta la representación gráfica de la estructura Original y la estructura Destino, vemos a clonar la estructura origen en la estructura destino.

K1 = K4

K2 = K5

V6 = frecuencia(K3)

```
TreeMap<String, TreeMap<String, ArrayList<String>>> datos;

public TreeMap<String, TreeMap<String, ArrayList<String, Integer>>> convertir() {
  private TreeMap<String, TreeMap<String, TreeMap<String, Integer>>> result;
  for (Entry<String, TreeMap<String, ArrayList<String>>> entry : this.datos.entrySet()) {
    TreeMap<String, TreeMap<String, Integer>> aux2 = new TreeMap<>();
    for (Entry<String, ArrayList<String>> entry2 : entry.getValue().entrySet()) {
      TreeMap<String, Integer> aux = new TreeMap<>();
      for (String palabra : entry2.getValue()) {
        Integer valor = aux.get(Palabra);
        aux.put(palabra, valor == null ? 1 : valor + 1);
      }
      aux.put(entry2.getKey(), aux);
    }
    result.put(entry.getKey(), aux2);
  }
  return result;
}
```

```
TreeMap<String, TreeMap<String, TreeMap<String, Integer>>> datosDestino;
AVLTree<Par<String, AVLTree<Par<String, AVLTree<Par<String, Integer>>>>>> datosOrigen;

public void convertir() {
  TreeMap<String, TreeMap<String, Integer>> aux;
  for (Par<String, AVLTree<Par<String, AVLTree<Par<String, Integer>>>>> parMap:
    datosOrigen) {
    datosDestino.put(parMap.getKey(), aux = new TreeMap<>());
    TreeMap<String, Integer> aux;
    for (Par<String, AVLTree<Par<String, Integer>>> parSecond: parMap.getValues()) {
      aux.put(parSecond.getKey(), aux2 = new TreeMap<>());
      for (Par<String, Integer> parThird: parSecond.getValue()) {
        aux2.put(parThird.getKey(), parThird.getValue());
      }
    }
  }
}
```