Algoritmos y Estructuras de Datos



Listas en Java

1

Listas

⊕UCU

- Secuencia de cero o más elementos de un tipo determinado
- Sucesión de elementos separados por comas:
 a₁, a₂, ..., a_n donde n >= 0 y a_i es del tipo de elemento
- n = 0 significa lista vacía
- a₁ es el primer elemento y a_n es el último
- a_i está en la posición i
- a_{i-1} precede a a_i , y a_i sucede a a_{i-1}
- los elementos pueden estar ordenados en forma lineal de acuerdo a sus posiciones en la lista

Algoritmos y Estructuras de Datos

2

2

⊕ UCU

Interface List

- Es la encargada de agrupar una colección de elementos uno a continuación del siguiente.
- Tiene la firma de todas las operaciones del TDA Lista visto en clase (y alguna más).

Algoritmos y Estructuras de Datos I

/ Estructuras de Datos I

Interface List • Similar a la que utilizamos en el curso – boolean isEmpty(); – boolean add(E e); – boolean remove(Object o); – int indexOf(Object o); – E remove(int index); – void clear();

Algoritmos y Estructuras de Datos

4

Interface List

⊕UCU

• Java -> PODEMOS VER EL CÓDIGO!!!

/**
* Returns the number of elements in this list. If this list contains
* more than <tt>Integer.MAX_VALUE</tt> elements, returns
* <tt>Integer.MAX_VALUE</tt> .
*
* @return the number of elements in this list
*/
<pre>int size();</pre>
/**
* Returns <tt>true</tt> if this list contains no elements.
*
* @return <tt>true</tt> if this list contains no elements
*/
boolean isEmpty();

Ilgoritmos y Estructuras de Datos I

5

ArrayList

⊕UCU

- Implementación de Lista con un array (Implementa **List**).
- Array dinámico -> se redimensiona según vaya necesitando más lugares en la lista.
- Beneficios ->
 - Rápido acceso a los elementos.
 - Otras bondades de los arrays.
- Problema -> redimensionar no es GRATIS!!!

Igoritmos y Estructuras de Datos I

Algoritmos y Estructuras de Datos I

7

8

LinkedList

- Implementación de una lista doblemente encadenada (implementa **List**).
- Beneficios ->
 - Inserciones y eliminaciones mucho más rápidas que ArrayList.
 - Manejo de memoria.

Desventajas -> búsquedas en *ArrayList* si conozco índices son mucho más rápidas (brinda la posibilidad de acceder directamente al Elemento).

Algoritmos y Estructuras de Datos I

⊕UCU

LinkedList



10

LinkedList

⊕UCU

```
* Links e as last element.

*/ Links e as last element.

*/ void linkLast(E e) {
    final NodecE> l = last;
    final NodecE> newNode = new Node<>(1, e, null);
    last = newNode;
    if (1 == null)
        first = newNode;
    sleet+;
    modCount++;
}

/**

* Inserts element e before non-null Node succ.

*void linkBefore(E e, NodecE> succ) {
    // assert succ |= null;
    final NodecE> pred = succ.prev;
    final NodecE> pred = succ.prev;
    final NodecE> newNode = new Node<>(pred, e, succ);
    succ.prev = newNode;
    if (pred == null)
        first = newNode;
    sleet
    pred.next = newNode;
    sleet
    pred.next = newNode;
}
```

11

11

LinkedList



```
/**
  * Appends the specified element to the end of this list.
  *
  * This method is equivalent to {@link #addLast}.
  *
  * @param e element to be appended to this list
  * @return {@code true} (as specified by {@link Collection#add})
  */
public boolean add(E e) {
  linkLast(e);
  return true;
```

Algoritmos y Estructuras de Datos I

12

Stack

⊕UCU

- La clase **Stack** es una lista en la que el acceso a sus elementos es de tipo LIFO (Last In - First Out, o último en entrar - primero en salir).
- Es la implementación del TDA Pila de Java
- Hereda de Vector (otra implementación de List, muy similar a ArrayList pero Synchronized).
- Stack hereda de Vector (es una List).

Algoritmos y Estructuras de Datos I

13

Stack

⊕UCU

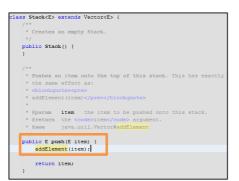
- Métodos
- public E push(E item)
 - public synchronized E pop()
 - public synchronized E peek()
 - public boolean empty()

Algoritmos y Estructuras de Datos I

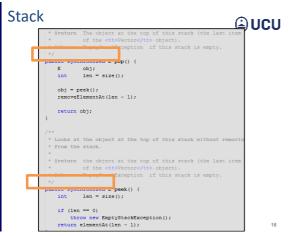
14

Stack





Algoritmos y Estructuras de Datos I



16

Resumiendo

⊕UCU

- Java utiliza varios de los algoritmos que utilizamos durante la primera mitad del curso.
- Estamos en condiciones de entender que es lo que está pasando ahí adentro (ya no es más una caja negra).

Algoritmos y Estructuras de Datos I

17

¿Preguntas?

⊕UCU

Igoritmos y Estructuras de Datos I