Tema: Estructuras de datos lineales

Tipos primitivos (int,double…)

Tipos de referencia 🡪 Extender el lenguaje 🡪Perro🡪Animal🡪object

Tipos de datos 🡪 operaciones: 🡪Int 🡪 +, -, x, /

* Byte 🡪&, >>, <<, |

TDA

🡪Tipos de Datos Abstractos

🡪Fundamento matemático

🡪Listas enlazadas: Add, AddFirst, AddLost, remove, remove first

🡪Arboles

🡪Grafos

🡪Set/conjuntos

🡪Maps/Diccionarios

🡪Colas/Pilas

Arrays

🡪Coleccion finita y bien definida de elementos de X tipo

🡪Cada elemento está contiguo uno al otro en la memoria

🡪Rapida lectura y escritura, pero lenta inserción de un elemento en desorden

🡪No puede decrecer ni crecer el arraylist

Int [ ] a = new int[ 8 ];

A [ 0 ]=10;

A [ 7 ]=20

Sys.out(a[ 3 ]);

Int x = a [ 0 ];

Como recorrer un array

For (int I = 0; i < a.length; i++){

Sys.out(a{[ i ]);

}

For (int i = a) {for each

Sys.out (i);

}

Matrices

🡪Array Multidimensional

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

🡪Filas \*y columnas \*\*

Int [ ][ ] m = new int [ 2 ]\*[ 2 ]\*\*;

🡪No todos los elementos están contiguos

Como imprimir una matriz

For (int i = 0; i< a. length; i++) {

For (int j = 0; j < a [ i ].length, j++) {

Sys.out(a[ i ][ j ]);

}

}

Buscar Hackererrand (mini programas de típicos errores)

Hackernoon

Medium

Listas Enlazadas

Estructuras de datos dinámicos

Crece o decrece por demanda

No está contiguo a memoria

Lista secuencial de elementos

Cada elemento es un nodo

X ||

Un nodo es un objeto compuesto por 2 elementos:

El valor que guarda o Dato

Referencia al siguiente elemento de la lista

First (foto)

La lista tiene una referencia al primer elemento

La referencia al primero debe protgerese

El ultimo nodo apunta siempre a null

Ventajas

Crece y decrece

Insercion en posiciones aleatorias es rápida

Desventajas

Lectura/búsqueda lenta

Package list

--- nodo

Public list

Class nodo {int dato;

Nodo next;

Public nodo(int dato){

This.dato = dato

This.next = null;}

}

Public class list {

Private nodo first;

Public list( ){

This.first = null;}

Public void addList(int e){

If (this.first == null){

This.first = new nodo(e);

}

Else {

Nodo current = this.first;

While(current.next != null){current = current.next;

}

Current.next = new nodo(e);

}

Main:

List l = new list( );

l.addLost(1);

l.addLost(2);

Buscar sobre generericts