public class Lista {

Nodo head;

public void insertarFinal(int valor) {

Nodo nuevo = new Nodo(valor);

if (head == null) {

head = nuevo;

} else {

Nodo actual = head;

O(n)

while (actual.siguiente != null) {

actual = actual.siguiente;

}

actual.siguiente = nuevo;

}

}

public Nodo invertir(Nodo nodo) {

Nodo anterior = null;

Nodo actual = nodo;

Nodo siguiente;

while (actual != null) {

siguiente = actual.siguiente;

O(n)

actual.siguiente = anterior;

anterior = actual;

actual = siguiente;

}

return anterior;

}

public Nodo restarListas(Nodo l1, Nodo l2) {

l1 = invertir(l1);

l2 = invertir(l2);

= O(n)

O(n) + O(m)

Nodo resultado = null;

Nodo actual = null;

int prestamo = 0;

while (l1 != null || l2 != null) {

int val1 = (l1 != null) ? l1.clave : 0;

int val2 = (l2 != null) ? l2.clave : 0;

int resta = val1 - val2 - prestamo;

if (resta < 0) {

resta += 10;

prestamo = 1;

} else {

prestamo = 0;

}

O(k)

Nodo nuevo = new Nodo(resta);

if (resultado == null) {

resultado = nuevo;

actual = nuevo;

} else {

actual.siguiente = nuevo;

actual = actual.siguiente;

}

if (l1 != null) l1 = l1.siguiente;

if (l2 != null) l2 = l2.siguiente;

}

O(k)

return invertir(resultado);

}

public void mostrar(Nodo nodo) {

while (nodo != null) {

System.out.print(nodo.clave + " -> ");

O(n)

nodo = nodo.siguiente;

}

System.out.println("null");

}

}