public class Juego {

private ArbolGeneral<Personaje> aux= null;

boolean encontre = false;

ListaGenerica<Personaje> camino = new ListaEnlazadaGenerica<Personaje>();

ListaGenerica<Personaje> caminoAux = new ListaEnlazadaGenerica<Personaje>();

public ListaGenerica<Personaje> encontrarPrincesa(ArbolGeneral<Personaje> arbol) {

lista.agregarFinal(arbol.getDatoRaiz());

encontrar(arbol);

return camino; ENCONTRAR A LA PRINCESA

}

private void encontrar(ArbolGeneral<Personaje> arbol) {

Personaje p = arbol.getDatoRaiz();

if (p.esPrincesa()) {

encontre = true;

camino = lista.clonar();

}

if (!encontre) {

ListaGenerica<ArbolGeneral<Personaje>> lHijos = arbol.getHijos();

lHijos.comenzar();

while (!lHijos.fin()) {

caminoAux = lHijos.proximo();

if (!aux.getDatoRaiz().esDragon()) {

lista.agregarFinal(aux.getDatoRaiz());

encontrar(aux);

lista.eliminarEn(lista.tamanio());

}

} } }}

package parcial;

//imports CAUDAL DE AGUA

public class RedAguaPotable {

public static double mnimoCaudal(ArbolGeneral<Integer> ag, double caudalEntrante) {

double minCaudal = caudalEntrante;

ListaGenerica<ArbolGeneral<Integer>> hijos = ag.getHijos();

if (!hijos.fin()) {

hijos.comenzar();

while (!hijos.fin()) {

minCaudal = Math.min(minCaudal, minimoCaudal(hijos.proximo(),caudalEntrante/hijos.tamanio()));

}

}

return minCaudal;

}]