

Práctica 4

```
public boolean existeCaminoSeguro() {
```

Encontrar un camino por el que llegar a un tesoro sin habitaciones con monstruos.

```
    public boolean existeCaminoSeguro() {  
        return existeCaminoRec(raiz);  
    }
```

```
    private boolean existeCaminoRec(NodoArbol nodo) {  
        if (nodo != null) {  
            if (nodo.getIzquierdo == null && nodo.getDerecho() == null &&  
                nodo.getDato().getValor() >= 0) {  
                return true;  
            } else if (nodo.getDato().getValor() < 0) {  
                if (existeCaminoRec(nodo.getIzquierdo)) {  
                    return true;  
                } else if (existeCaminoRec(nodo.getDerecho)) {  
                    return true;  
                }  
            }  
        }  
        return false;  
    }
```

Práctica 5

public int tamañoMayorGrupo()

Devuelve el número de integrantes del grupo más grande

```
public int tamañoMayorGrupo() {  
    boolean[] visitados = new boolean[red.getNumVertices];  
    int max = 0;  
    for (int i = 0; i < red.getNumVertices; i++) {  
        if (!visitados[i]) {  
            boolean[] grupo = red.profundidadDesdeVertice(i);  
            int numtrue = 0;  
            for (int j = 0; j < red.getNumVertices(); j++) {  
                if (grupo[j]) {  
                    visitados[j] = true;  
                    numtrue++;  
                }  
            }  
            if (numtrue > max) {  
                max = numtrue;  
            }  
        }  
    }  
    return max;  
}
```