**Report**

# 1.

O construtor do iterador está mal, a raiz é pushed para a stack sem verificar se a tree é emtpy

É impossível cobrir todos os branches do método equalTrees. O branch “one == other” já é coberto no método equals e que caso aconteça, o método equalTrees não é chamado. O branch e “one == null || other == null” também pois, o one aqui seria o this, que não poderia ser null se não ao tentar chamar o método equals seria lançada uma NullPointerException, e o other também nunca poderia ser null pois a condição “other instanceof NTree” não se verificaria e o método equalTrees não seria chamado.

É também impossível cobrir todos os branches do método proposePosition. No if “if (children[i].data.compareTo(elem)<0)” é impossível cobrir o branch em que esta condição não se verifica, isto porque todas as outras condições (ser igual ou maior do que 0) já foram verificadas em partes anteriores do código. Portanto para obter cobertura total, o código teria de ser escrito com if, else if e else, visto que no fim não há outra alternativa a “children[i].data.compareTo(elem)<0”.

# 2.

TR(EC) = {[0,1], [0,2], [2,3], [2,4]}

TR(EPC) = TR(EC) ∪ {[0,2,3], [0,2,4]}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nome do teste | Valores  (this,other) | Resultado esperado | Caminho | Requerimentos cobertos |
| testArrayNTreeEqualsSameRef | this e other são a mesma instância | true | [0,1] | [0,1] |
| testArrayNTreeEqualsOtherNotNTree | ([1],1) | false | [0,2,3] | [0,2,3] |
| testArrayNTreeEqualsEqualNTreesDiffRef | ([1],[1]) | true | [0,2,4] | [0,2,4] |

\*Para a representação dos valores foi usada a representação toString dos objetos ArrayNTree

# 3.

# 4.

Parâmetros de input: this, other

Características:

* isNTree: Yes/No
* Size: 0,1, 2+
* childrenCapacity: 0, 1, 2+