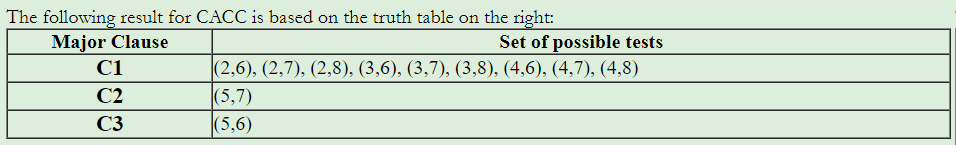
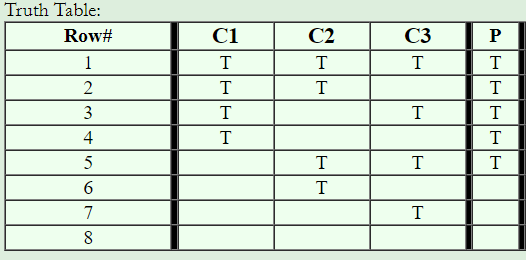
**EQUALS**

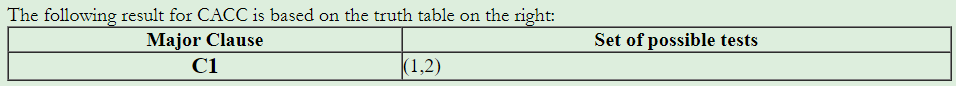
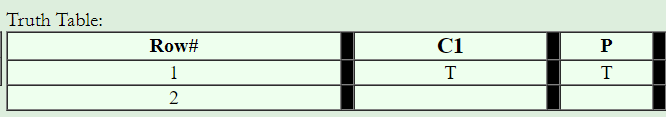
**P1:** C1 || C2 && C3  
C1: this == other  
C2: other instanceof NTree  
C3: equalTrees(this, ((NTree<T>) other))

Testes impossíveis:   
2 -   
3 -  
4 -  
7 -

Testes utilizados para cumprir o C1: (8), pois os testes 2, 3, 4 e 7 são impossíveis de serem realizados.  
Testes utilizados para cumprir o C2: (5), pois o teste 7 é impossível de ser realizado.  
Testes utilizados para cumprir o C3: (5,6)

**EQUALTREES**

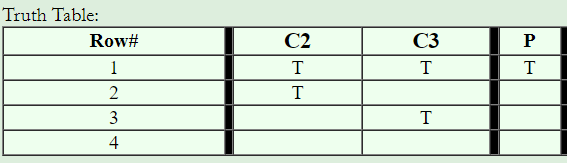
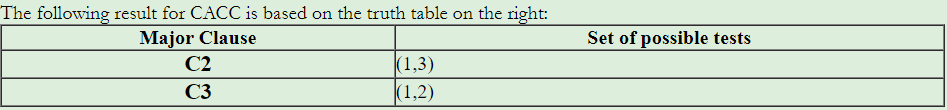
**P1**: C1  
C1: one == other



Testes impossíveis:   
1 -

Testes utilizados para cumprir o C1: (2), pois o teste 1 é impossível de ser realizado.

**P2**: C2 && C3  
C2: one != null  
C3: other != null

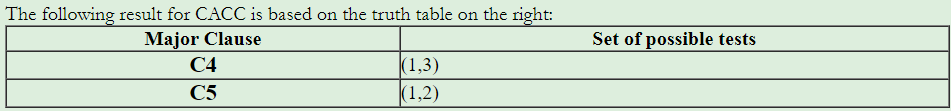
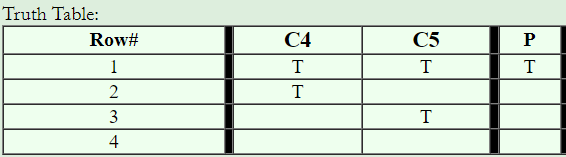
Testes impossíveis:   
2 -  
3 -  
4

Testes utilizados para cumprir o C2: (1), pois o teste 3 é impossível de ser realizado.

pois é impossível realizar o testar quando C3 é falso, i.e, quando other == null, como já foi explicado anteriormente.  
Testes utilizados para cumprir o C3: (1), pois o teste 2 é impossível de ser realizado.

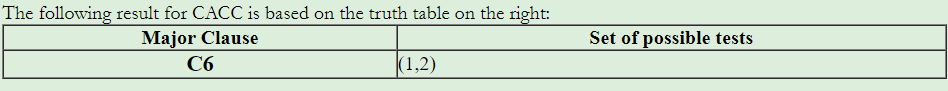
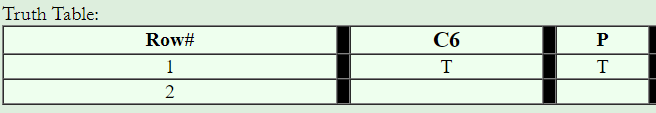
pois é impossível realizar o testar quando C2 é falso, i.e, quando one == null, como já foi explicado anteriormente.

**P3**: C4 && C5  
C4: it1.hasNext()  
C5: it2.hasNext()



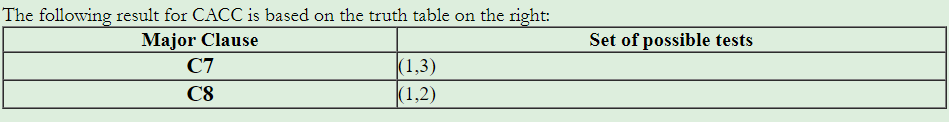
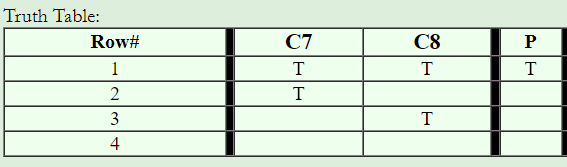
Testes utilizados para cumprir o C4: (1,3).  
Testes utilizados para cumprir o C5: (1,2).

**P4**: C6  
C6: !it1.next().equals(it2.next())



Testes utilizados para cumprir o C6: (1,2).

**P5**: C7 && C8  
C7: !it1.hasNext()  
C8: !it2.hasNext()



Testes impossíveis:   
4 -

Testes utilizados para cumprir o C7: (1,3).  
Testes utilizados para cumprir o C8: (1,2).