



## ALD GRP 5 Explanation TraverseTree

### countWordsInSubtree

1. counter = 0
2. wenn w leer soll 0 zurückgegeben werden ansonsten wird der counter um 1 erhöht
3. dann prüfe ob es ein linkes kind gibt
  - 3.1 erhöhe den counter um den return Wert der Rekursion  
für das linke Element
4. wenn der Rechte nicht null ist,
  - 4.1 erhöhe counter um den return Wert der Rekursion  
für das rechte Element
5. return counter

In Worten:      So wird im ersten Durchlauf bis in die unterste Ebene des Baums gegangen und der counter für jede Stufe erhöht. gibt es keine Elemente darunter (links + rechts) geht man aus der "kleinsten Schachtel"(=scope) der Rekursion heraus und prüft den "sibling" für das vorige Element  
Gibt es weder linkes und rechtes ein "child", wird in die nächstgrößere Schachtel zurückgegangen, und der Baum wird wieder bis zur untersten Ebene durchsucht

#### getWordsWithPrefix

1. erstelle temporäres Wort = root;  
    erstelle einen counter = 0;  
    erstelle ein String Set;  
    erstelle einen Wort Stack;
2. Wenn Wort null ist , return null
3. Wenn das Wort den Prefix enthält, adde das Wort dem Set und pushe auf Stack  
    Stack darf für nächsten Schritt nicht leer sein
4. Endlosschleife
  - 4.1 Wenn der Stack nicht leer ist, Wort = Stack.pop
    - 4.1.1 ansonsten break
  - 4.2 Wenn das linke Wort nicht null ist, pushe das linke auf den Stack
  - 4.3 Wenn das rechte Wort nicht null ist, pushe das rechte auf den Stack
  - 4.4 Wenn das Wort(4.1) den präfix enthält, füge es dem Set hinzu
5. Wenn Schleife unterbrochen --> return String Set