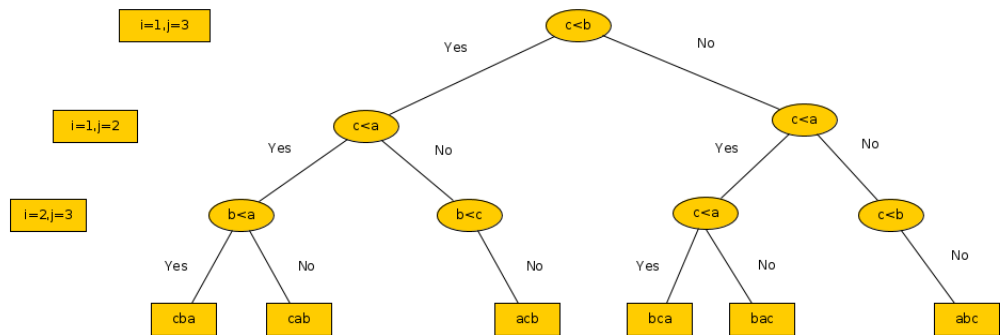


# Aufgabenblatt 3 - Aufgabe 4

16. November 2014

- (a) Zwei Yes-Zweige entfallen, da jeweils bereits vorher der selbe Vergleich durchgeführt wurde. Aufgrund des Bubble-Sort Algorithmus wird der Vergleich dennoch ein zweites mal berechnet.



- (b) Damit BUBBLE-SORT im Best-Case in  $O(n)$  liegt, kann man den Algorithmus so verändern, dass, wenn in einem Durchlauf der äußeren Schleife keine Elemente vertauscht werden, der Algorithmus abbricht. Dies liefert eine Laufzeit in  $O(n)$  bei einer bereits sortierten Eingabe.

```

(c) BUBBLESORT(A)
    exchanged=true
    for i=1 to A.length-1
        if exchanged==true
            exchanged=false
            for j= A.length downto i+1
                if A[j]<A[j-1]
                    exchange A[j] with A[j-1]
                    exchanged=true
            else leavefor
    
```

- (d) Bei jedem Durchlauf der äußeren Schleife, wird nun ein der Vergleich, ob `exchanged true` ist und die Zuweisung des Wertes `false` zu `exchanged` mehr durchgeführt. Weiterhin wird in der inneren Schleife, ebenfalls eine Wertzuweisung mehr durchgeführt.

Die Laufzeit des Algorithmus wird dadurch nur um einen konstanten Faktor erhöht, während sich die Best-Case Laufzeit auf einen Durchlauf durch das Array also  $O(n)$  reduziert wird. Somit lohnt sich die Anpassung.

- (e) Der Baum unterscheidet sich im Zweig ganz rechts. Wenn während  $i=1$  keine Vertauschung durchgeführt wird, bricht der Algorithmus vor  $i=2$  ab.

