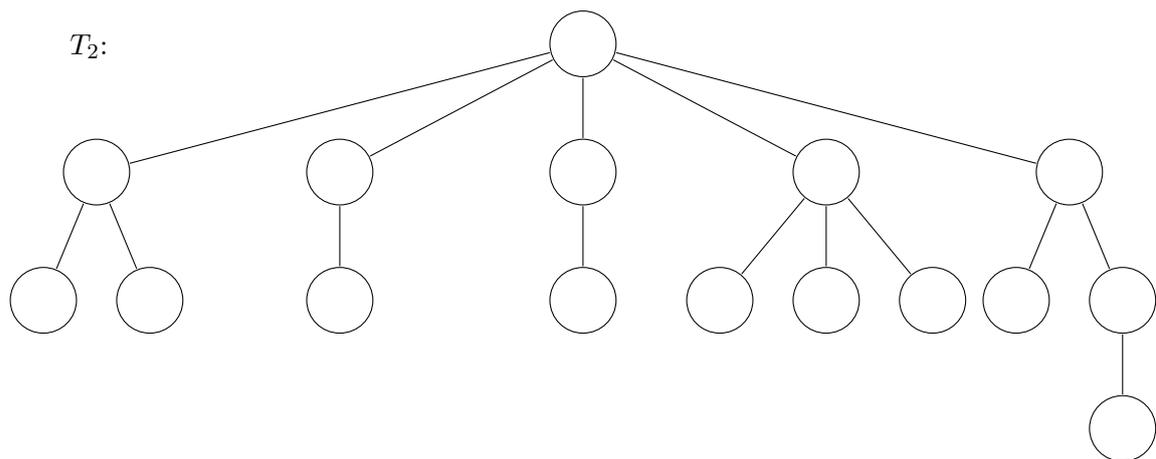
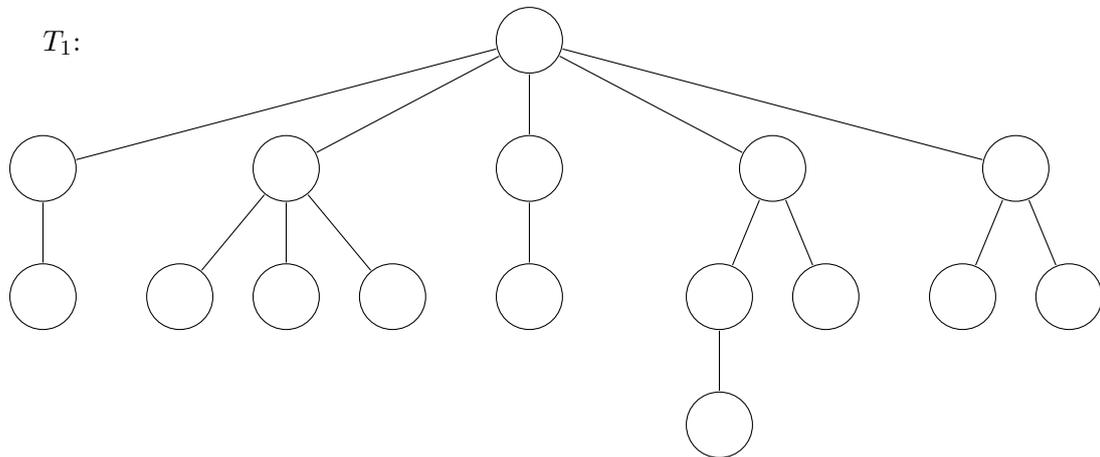


Präsenzaufgabe 3.1

Untersuchen Sie die beiden folgenden Bäume T_1 und T_2 unter Verwendung des Algorithmus aus der Vorlesung auf Isomorphie:



Geben Sie zu jedem inneren Knoten in T_1 und T_2 das zugehörige berechnete k -Tupel an, wobei k die Anzahl seiner Kinder sei.

Präsenzaufgabe 3.2

Sortieren Sie das folgende Array mit Hilfe von Heapsort:

A

| | | | | | |
|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 |
| 87 | 22 | 30 | 83 | 96 | 99 |

- Geben Sie den Baum nach jedem Schleifendurchlauf innerhalb der Prozedur BUILDHEAP an, falls es sich geändert hat.
- Geben Sie den von HEAPIFY betrachteten Teilbaum nach jedem Durchlauf der Hauptschleife an.

Ein weiterer Sortieralgorithmus ist BUBBLESORT. Er arbeitet wie folgt:

```
Prozedur BUBBLESORT(A) :  
  begin  
    for  $j = n - 1$  to 1 step -1 do  
      for  $i = 1$  to  $j$  step 1 do  
        if  $A[i + 1] < A[i]$  then tausche  $A[i]$  mit  $A[i + 1]$  end  
      end  
    end  
  end  
end
```

Präsenzaufgabe 3.3

Ein Sortieralgorithmus heißt *stabil*, wenn sich die relative Position zweier gleicher Elemente nach der Sortierung im Vergleich zur Ausgangsposition in der unsortierten Liste nicht geändert hat.

Begründen Sie welche der folgenden Sortieralgorithmen stabil sind und welche nicht.

- a) Bucketsort
- b) Heapsort
- c) Bubblesort