JOHANNES KEPLER
UNIVERSITY LINZ

Praktische Informatik 2

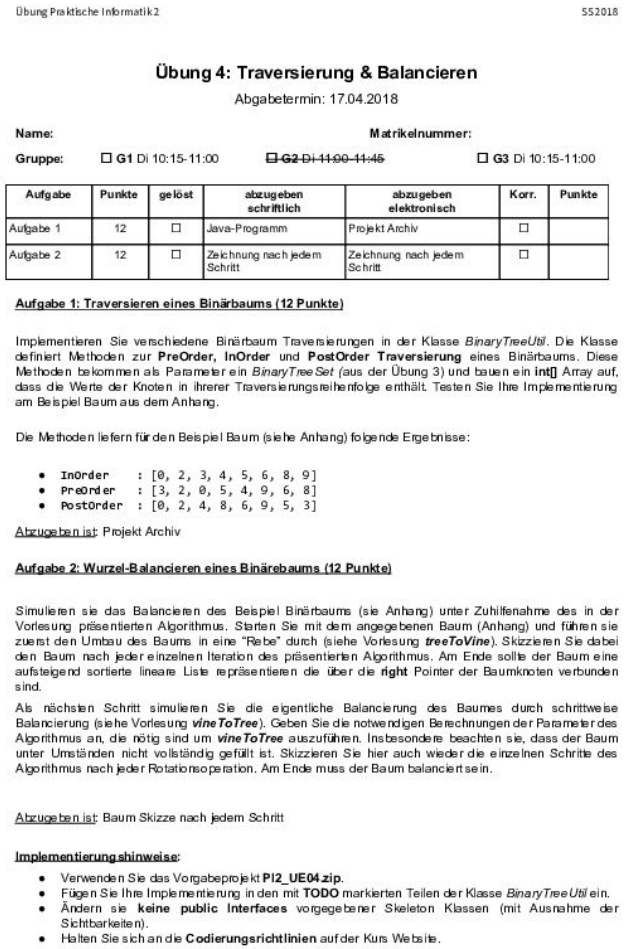
Übung, 2018S
Institut für System Software (SSW)
DI Eisl & DI Leopoldseder

JOHANNES KEPLER
UNIVERSITY LINZ

Ablauf

1. Fragen Übung 3
2. Fragen VO (Traversieren & Balancieren)
3. Übung 4
    a. Implementierung Traversierung

Übung 4 - Traversieren & Balancieren



Übung Praktische Informatik 2 SS2018

Übung 4: Traversierung & Balancieren

Abgabetermin: 17.04.2018

Name: **Matrikelnummer:**

Gruppe: ☐ **G1** Di 10:15-11:00 ☐ ~~**G2** Di 11:00-11:45~~ ☐ **G3** Di 10:15-11:00

Aufgabe	Punkte	gelöst	abzugeben schriftlich	abzugeben elektronisch	Korr.	Punkte
Aufgabe 1	12	☐	Java-Programm	Projekt Archiv	☐	
Aufgabe 2	12	☐	Zeichnung nach jedem Schritt	Zeichnung nach jedem Schritt	☐	

Aufgabe 1: Traversieren eines Binärbaums (12 Punkte)

Implementieren Sie verschiedene Binärbaum Traversierungen in der Klasse *BinaryTreeUtil*. Die Klasse definiert Methoden zur **PreOrder, InOrder** und **PostOrder Traversierung** eines Binärbaums. Diese Methoden bekommen als Parameter ein *BinaryTreeSet (*aus der Übung 3) und bauen ein **int[]** Array auf, dass die Werte der Knoten in ihrerer Traversierungsreihenfolge enthält. Testen Sie Ihre Implementierung am Beispiel Baum aus dem Anhang.

Die Methoden liefern für den Beispiel Baum (siehe Anhang) folgende Ergebnisse:

- `InOrder : [0, 2, 3, 4, 5, 6, 8, 9]`
- `PreOrder : [3, 2, 0, 5, 4, 9, 6, 8]`
- `PostOrder : [0, 2, 4, 8, 6, 9, 5, 3]`

Abzugeben ist: Projekt Archiv

Aufgabe 2: Wurzel-Balancieren eines Binärebaums (12 Punkte)

Simulieren sie das Balancieren des Beispiel Binärbaums (sie Anhang) unter Zuhilfenahme des in der Vorlesung präsentierten Algorithmus. Starten Sie mit dem angegebenen Baum (Anhang) und führen sie zuerst den Umbau des Baums in eine "Rebe" durch (siehe Vorlesung ***treeToVine***). Skizzieren Sie dabei den Baum nach jeder einzelnen Iteration des präsentierten Algorithmus. Am Ende sollte der Baum eine aufsteigend sortierte lineare Liste repräsentieren die über die **right** Pointer der Baumknoten verbunden sind.

Als nächsten Schritt simulieren Sie die eigentliche Balancierung des Baumes durch schrittweise Balancierung (siehe Vorlesung ***vineToTree***). Geben Sie die notwendigen Berechnungen der Parameter des Algorithmus an, die nötig sind um ***vineToTree*** auszuführen. Insbesondere beachten sie, dass der Baum unter Umständen nicht vollständig gefüllt ist. Skizzieren Sie hier auch wieder die einzelnen Schritte des Algorithmus nach jeder Rotationsoperation. Am Ende muss der Baum balanciert sein.

Abzugeben ist: Baum Skizze nach jedem Schritt

Implementierungshinweise:

- Verwenden Sie das Vorgabeprojekt **PI2_UE04.zip**.
- Fügen Sie Ihre Implementierung in den mit **TODO** markierten Teilen der Klasse *BinaryTreeUtil* ein.
- Ändern sie **keine public Interfaces** vorgegebener Skeleton Klassen (mit Ausnahme der Sichtbarkeiten).
- Halten Sie sich an die **Codierungsrichtlinien** auf der Kurs Website.

- Abgabetermin: 17.04.2018 08:30
- Vorgabe Projekt
- ANT zum Builden
- **TODO** markierte Methoden in Klasse ***BinaryTreeUtil.java*** implementieren

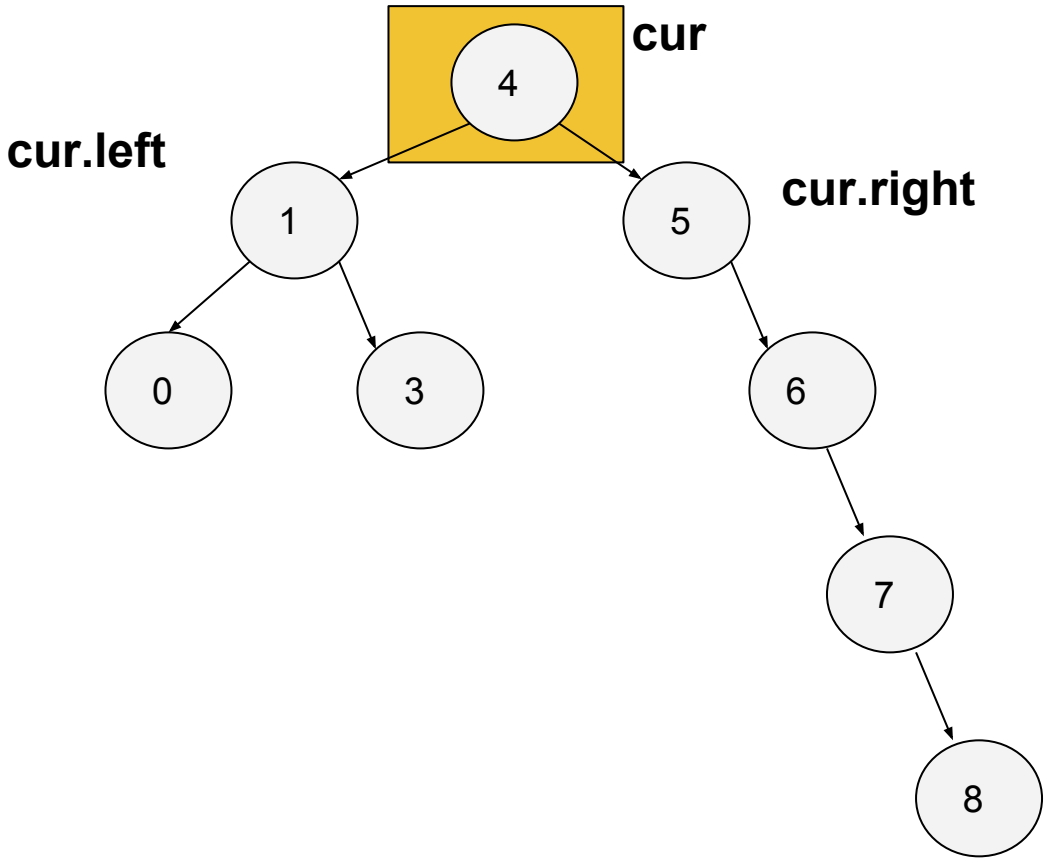
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[ ]



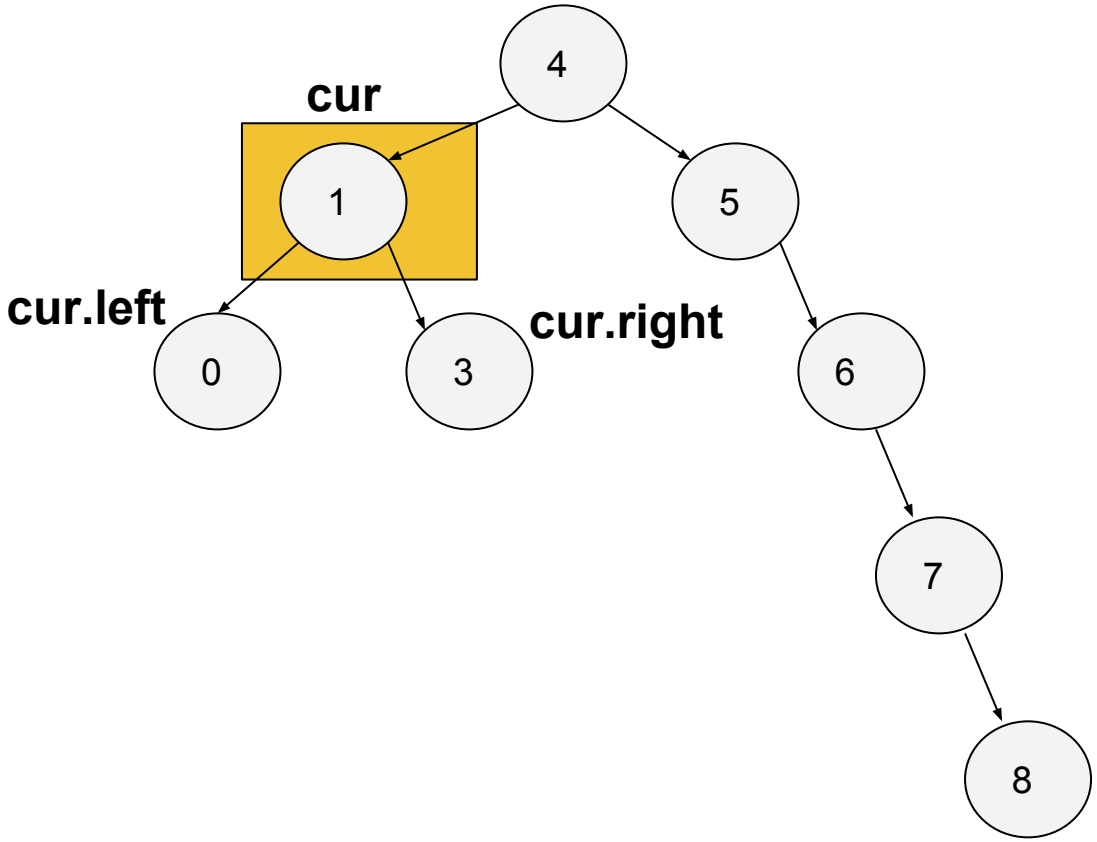
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[ ]



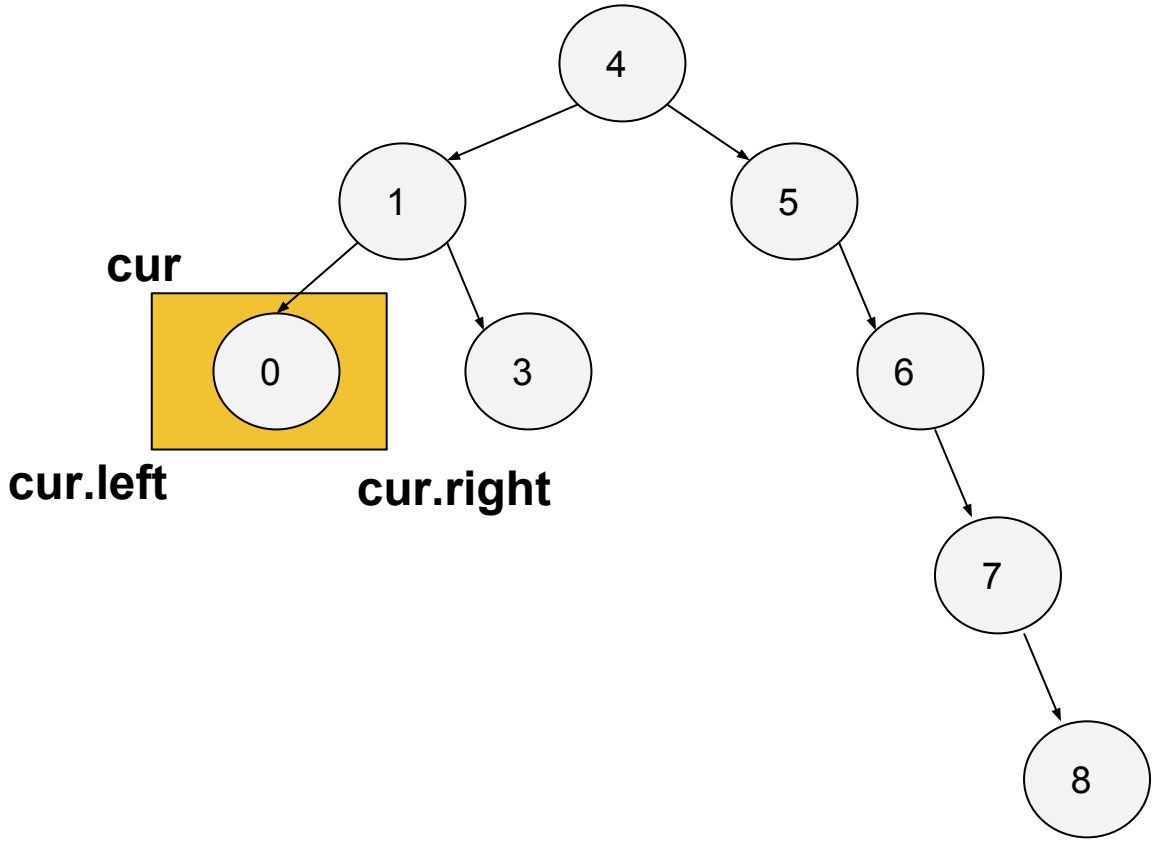
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[ ]



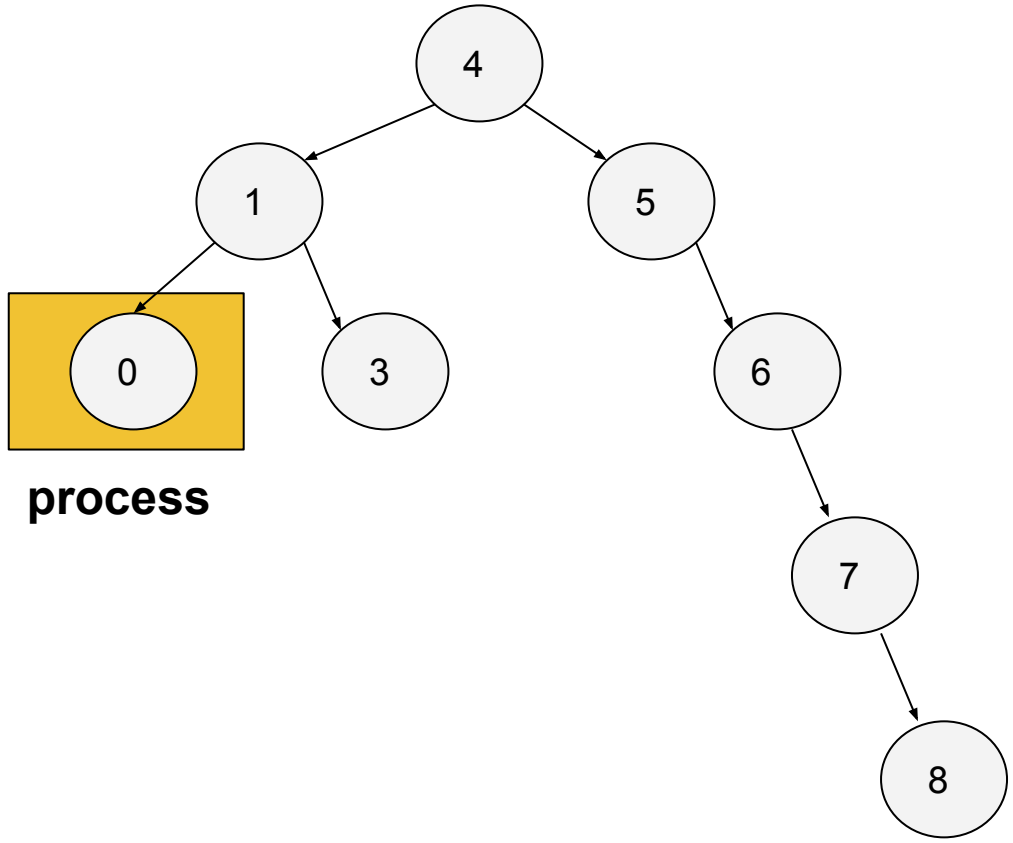
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0]



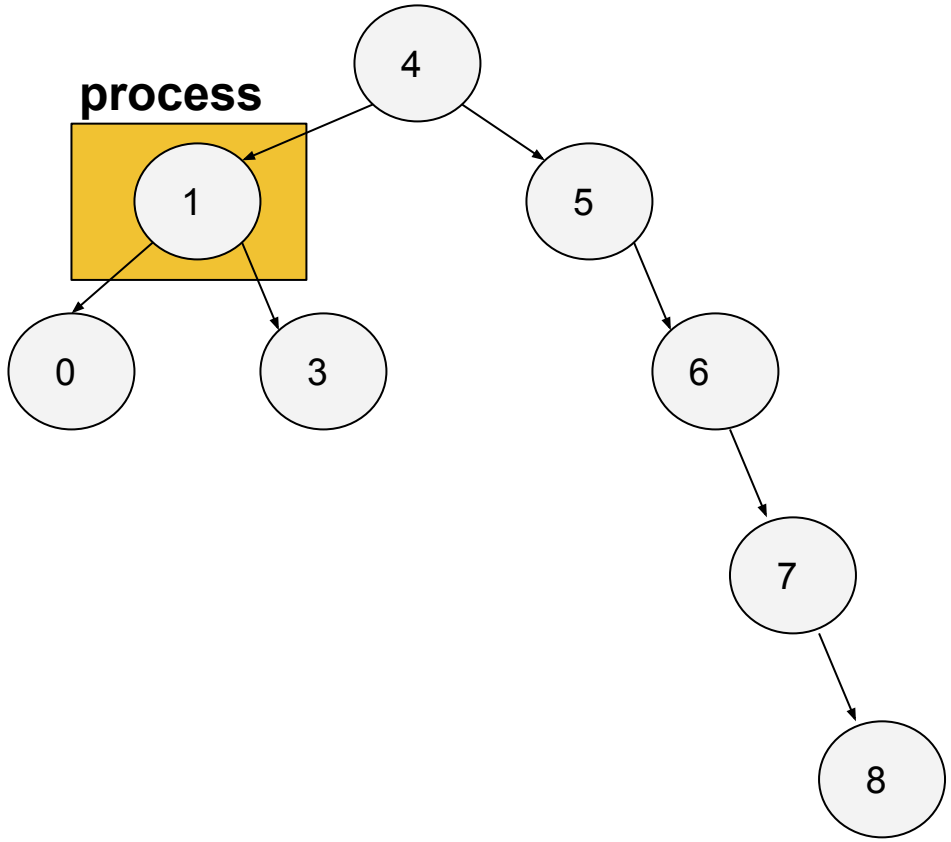
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1]



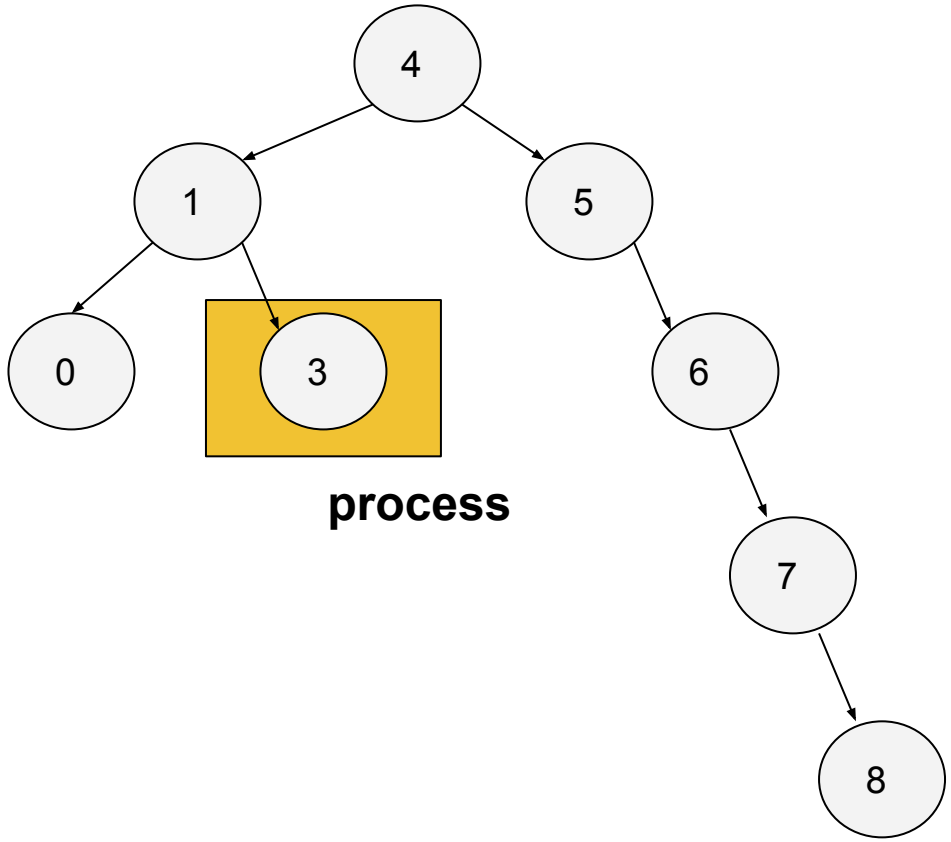
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1 - 3]



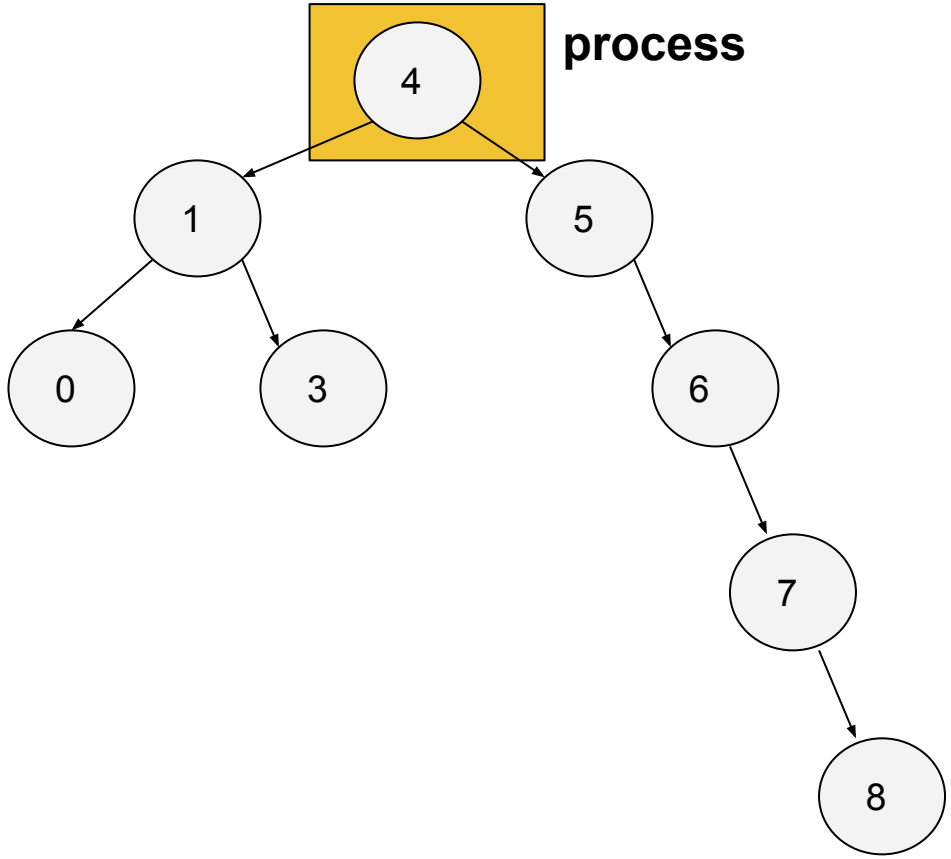
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1 - 3 - 4]



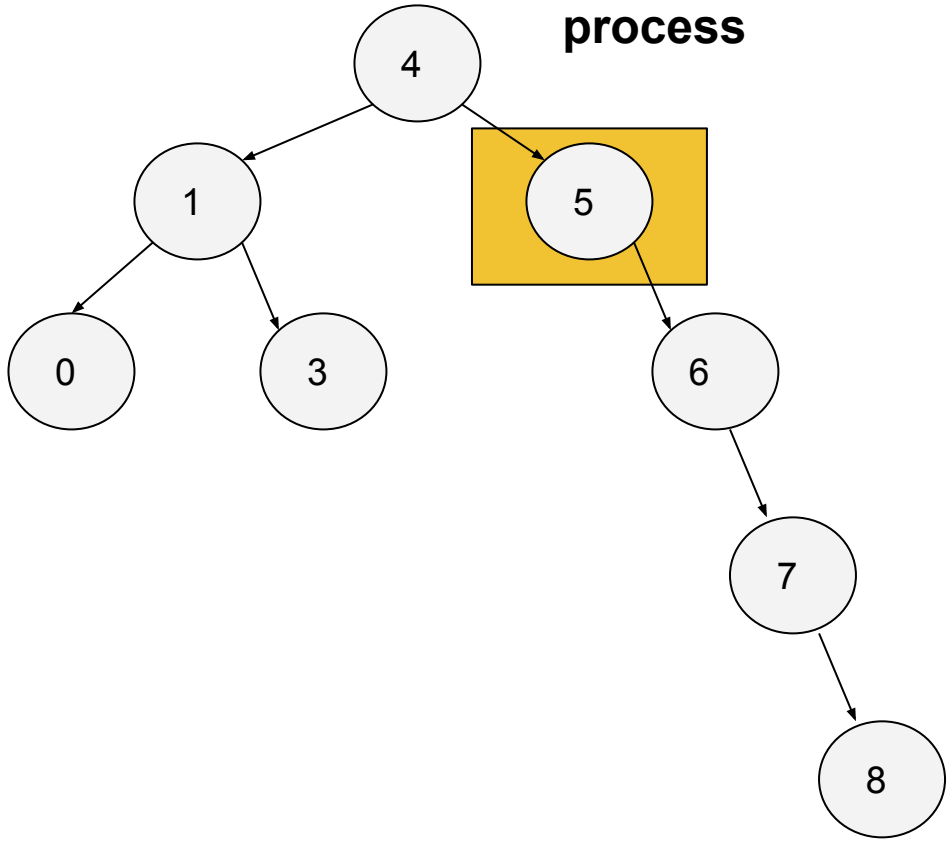
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1 - 3 - 4 - 5]



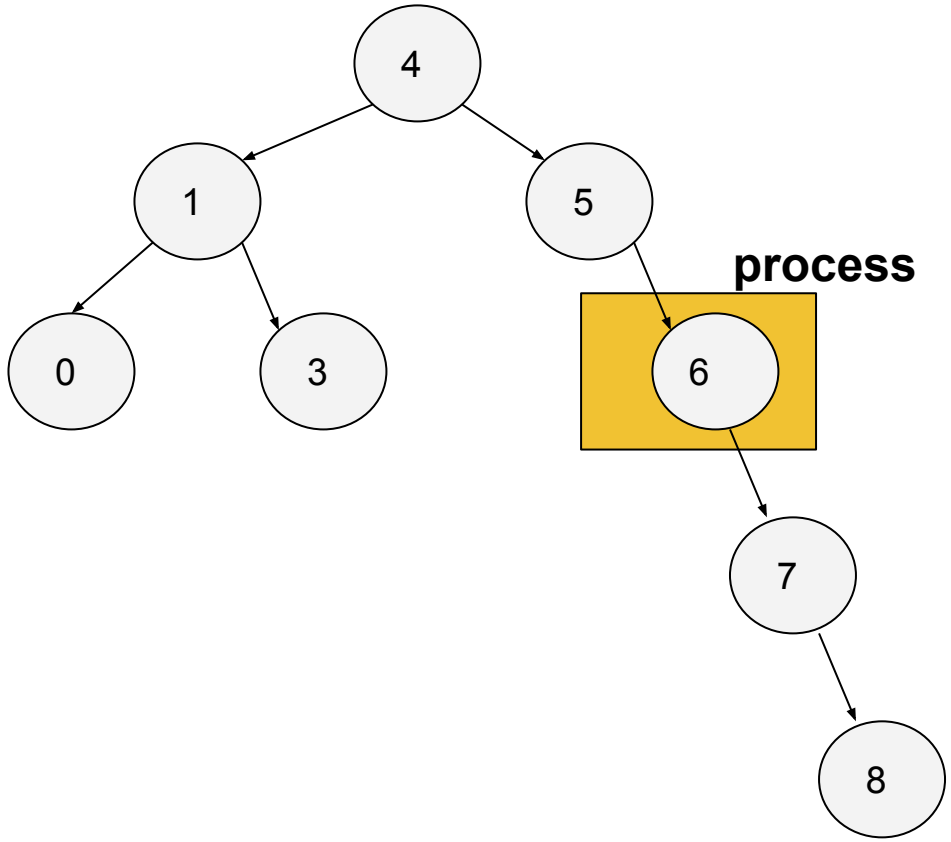
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1 - 3 - 4 - 5 - 6]



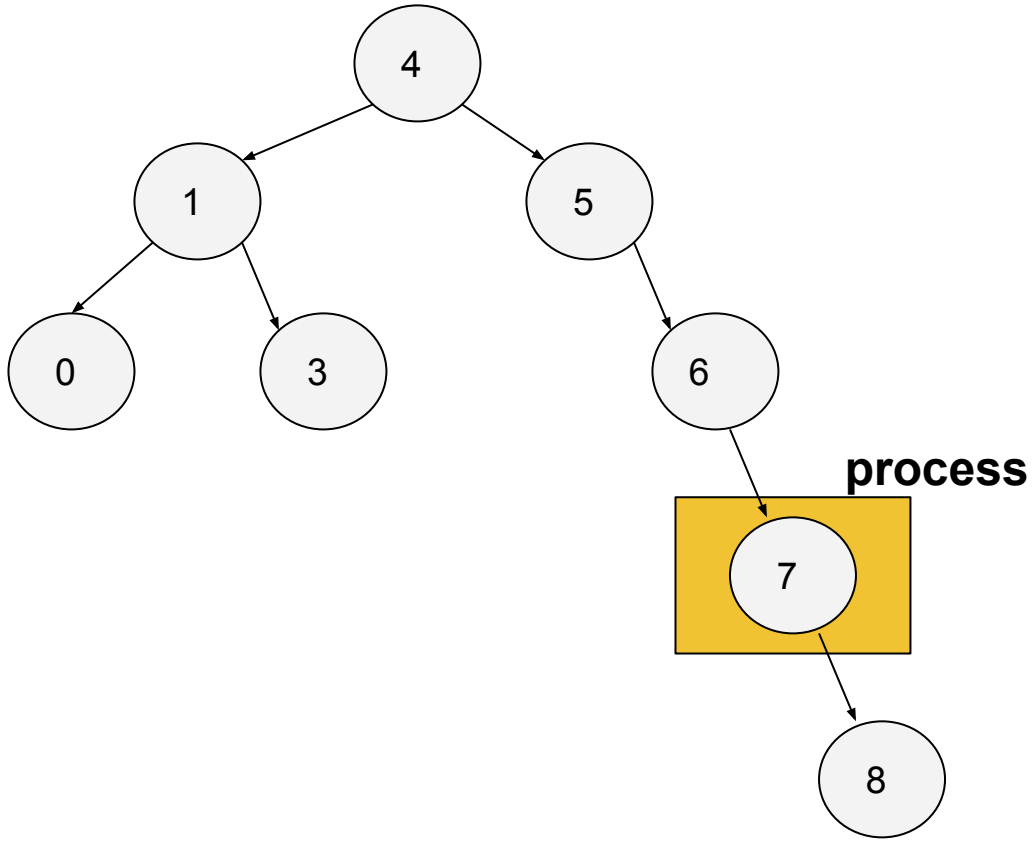
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1 - 3 - 4 - 5 - 6 - 7]



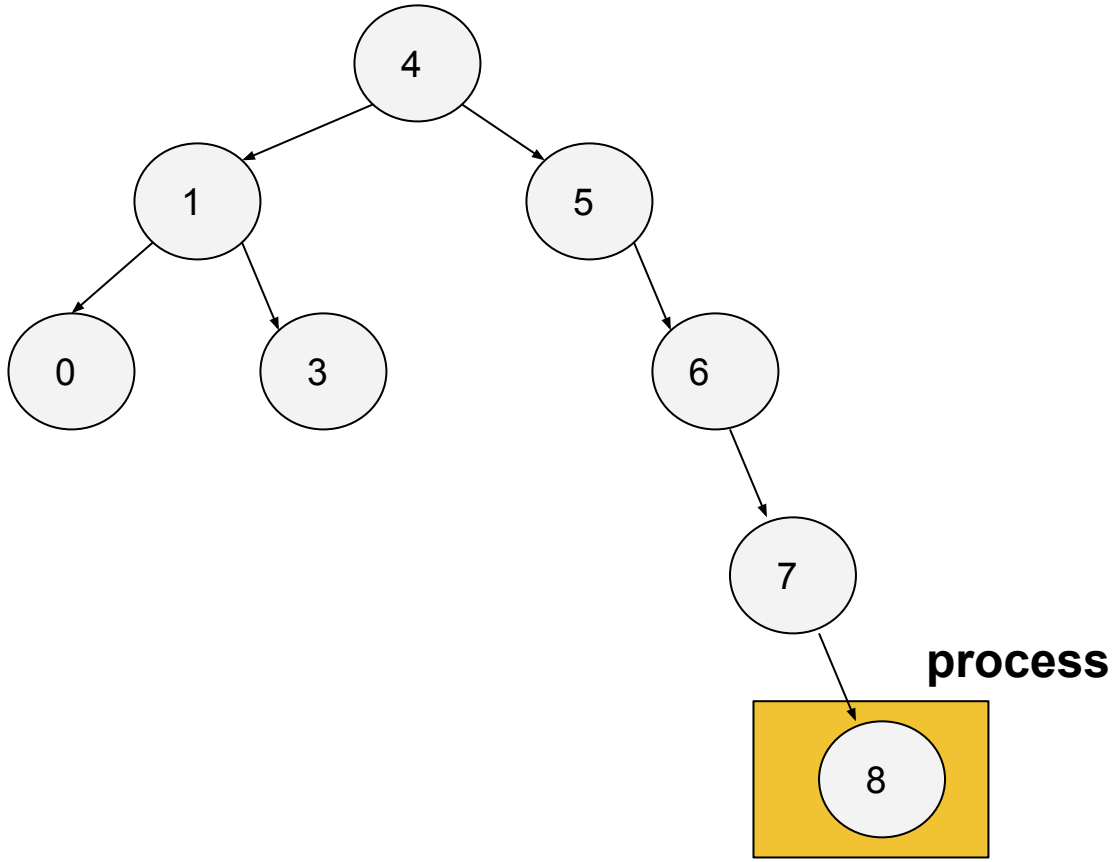
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1 - 3 - 4 - 5 - 6 - 7 - 8]



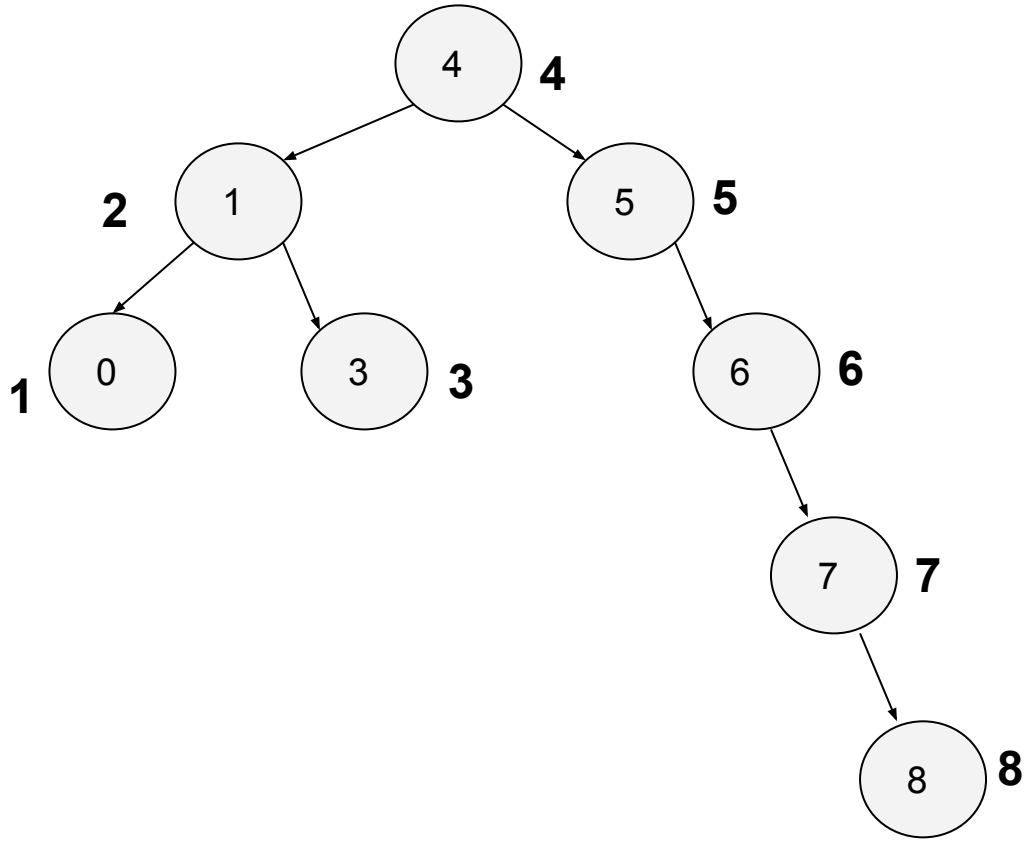
Übung 4 - Traversieren & Balancieren

InOrder

1. process cur.left
2. process cur
3. process cur.right

Ausgabe

[0 - 1 - 3 - 4 - 5 - 6 - 7 - 8]



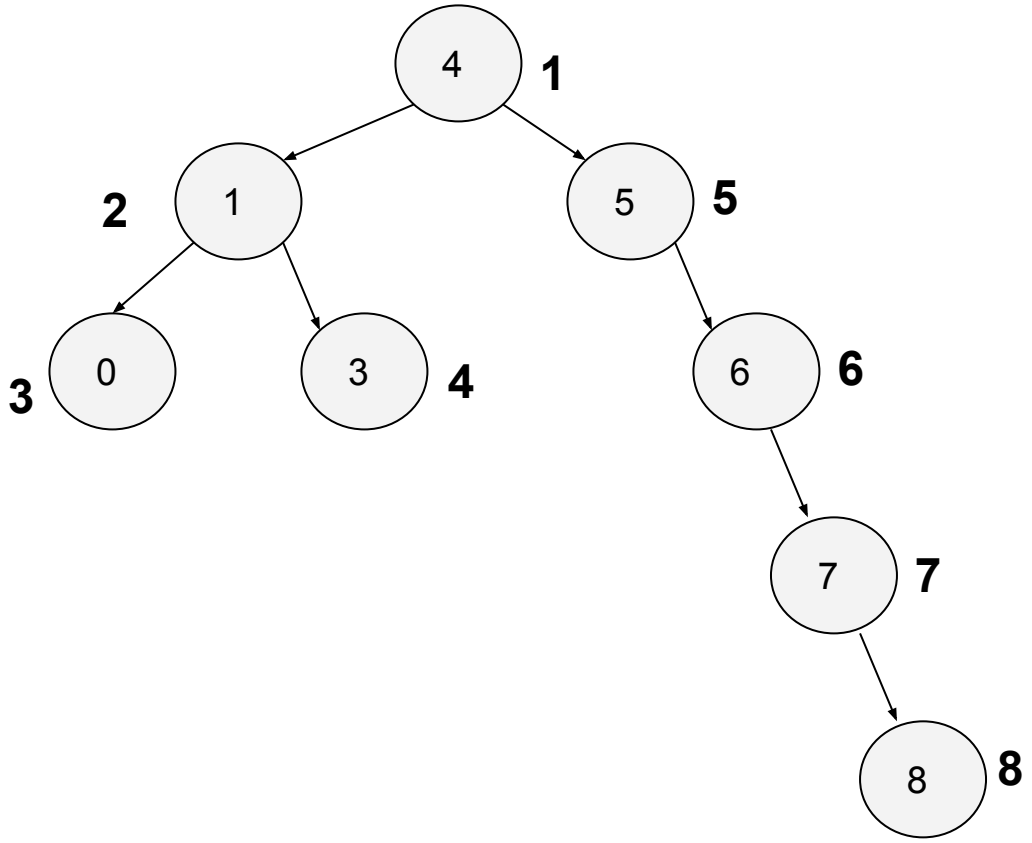
Übung 4 - Traversieren & Balancieren

PreOrder

1. process cur
2. process cur.left
3. process cur.right

PreOrder

[4 - 1 - 0 - 3 - 5 - 6 - 7 - 8]



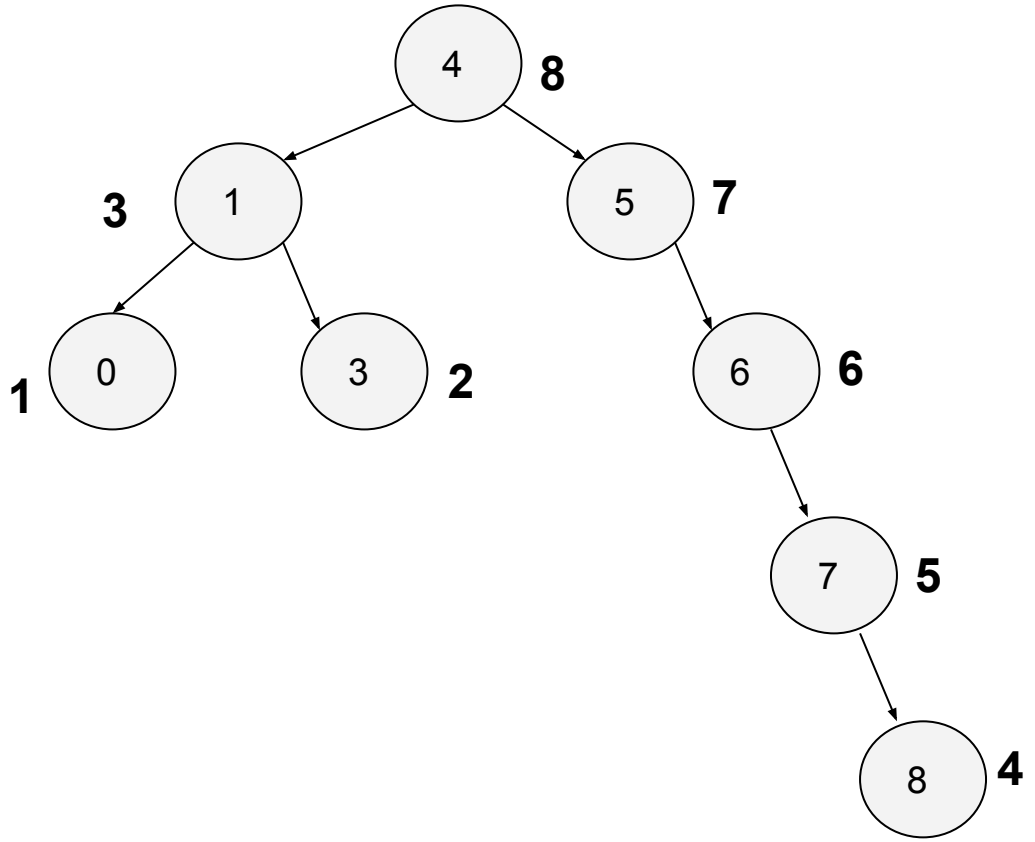
Übung 4 - Traversieren & Balancieren

PostOrder

1. process cur.left
2. process cur.right
3. process cur

PostOrder

[0 - 3 - 1 - 8 - 7 - 6 - 5 - 4]



JOHANNES KEPLER
UNIVERSITY LINZ

Danke

JOHANNES KEPLER
UNIVERSITY LINZ
Altenberger Str. 69
4040 Linz, Austria
www.jku.at