Informatik I: Einführung in die Programmierung 9. Bäume

Albert-Ludwigs-Universität Freiburg

Peter Thiemann

27. November 2018



Der Baum

Der Baum

Definition Terminologie Beispiele

Binärbäume

Suchbäume

Bäume in der Informatik

Bäume sind in der Informatik allgegenwärtig.



Der Baum

Definition Terminologie Beispiele

Binärbäume Suchbäume

_

Bäume in der Informatik

- Bäume sind in der Informatik allgegenwärtig.
- Gezeichnet werden sie meistens mit der Wurzel nach oben!



Der Baum

Definition Terminolog Beispiele

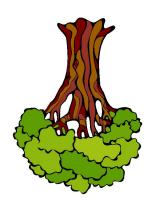
Binärbäume Suchbäume

Zusammenfassung

27. November 2018 P. Thiemann – Info I 4 / 40

Bäume in der Informatik

- Bäume sind in der Informatik allgegenwärtig.
- Gezeichnet werden sie meistens mit der Wurzel nach oben!



Der Baum

Definition Terminologi Beispiele

Binärbäume Suchbäume



Induktive Definition:

Der Baum

Definition Terminologie

Beispiele

Binärbäume

Suchbäume



- Induktive Definition:
 - Der leere Baum ist ein Baum.

Der Baum

Definition

Terminologie Beispiele

Binärbäume

Suchbäume



Induktive Definition:

- Der leere Baum ist ein Baum.
- Wenn $t_1, ..., t_n$, $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in t_1, \ldots, t_n vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen t_1, \ldots, t_n ein Baum.

Der Baum

Definition

fassung



.

- Induktive Definition:
 - Der leere Baum ist ein Baum.
 - Wenn t_1, \ldots, t_n , $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in t_1, \ldots, t_n vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen t_1, \ldots, t_n ein Baum.
 - Nichts sonst ist ein Baum.

Der Baum

Definition

Poissiolo

Binärbäume

7......



Induktive Definition:

- Der leere Baum ist ein Baum.
- Wenn $t_1, ..., t_n$, $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in $t_1, ..., t_n$ vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen $t_1, ..., t_n$ ein Baum.
- Nichts sonst ist ein Baum.
- Beispiel:

Der Baum

Terminolo

Beispiele

Binarbaume

Zusamme

fassung





- Induktive Definition:
 - Der leere Baum ist ein Baum.
 - Wenn $t_1, ..., t_n$, $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in $t_1, ..., t_n$ vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen $t_1, ..., t_n$ ein Baum.
 - Nichts sonst ist ein Baum.
 - Beispiel:



Der Baum

Terminolo

Beispiele

Binarbaume

Zusammer





- Induktive Definition:
 - Der leere Baum ist ein Baum.
 - Wenn $t_1, ..., t_n$, $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in $t_1, ..., t_n$ vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen $t_1, ..., t_n$ ein Baum.
 - Nichts sonst ist ein Baum.
 - Beispiel:





Der Baum

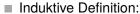
Terminolo

Binärhäume

Cacindaaiii







- Der leere Baum ist ein Baum.
- Wenn t_1, \ldots, t_n , $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in t_1, \ldots, t_n vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen t_1, \ldots, t_n ein Baum.
- Nichts sonst ist ein Baum.
- Beispiel:







Der Baum

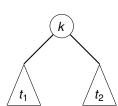
Terminolog

Binärbäume





- Induktive Definition:
 - Der leere Baum ist ein Baum.
 - Wenn $t_1, ..., t_n$, $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in $t_1, ..., t_n$ vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen $t_1, ..., t_n$ ein Baum.
 - Nichts sonst ist ein Baum.
 - Beispiel:



Der Baum

Terminolog

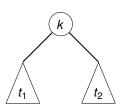
Binärbäume

Suchbaume





- Induktive Definition:
 - Der leere Baum ist ein Baum.
 - Wenn t_1, \ldots, t_n , $n \ge 0$ disjunkte Bäume sind und k ein Knoten, der nicht in t_1, \ldots, t_n vorkommt, dann ist auch die Struktur bestehend aus der Wurzel k mit zugeordneten Teilbäumen t_1, \ldots, t_n ein Baum.
 - Nichts sonst ist ein Baum.
 - Beispiel:



Beachte: Bäume können auch anders definiert werden und können auch eine andere Gestalt haben (z.B. ungewurzelt). Der Baum

Definition Terminologie

Binärbäume

Guoribuurn

Terminologie I



Alle Knoten, denen keine Teilbäume zugeordnet sind, heißen Blätter.

Der Baum Definition

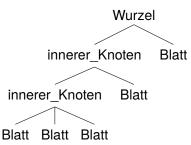
Terminologie Beispiele

Binärbäume

Suchbäume

Terminologie I

- (E)
- Alle Knoten, denen keine Teilbäume zugeordnet sind, heißen Blätter.
- Knoten, die keine Blätter sind, heißen innere Knoten.



Der Baum

Terminologie Beispiele

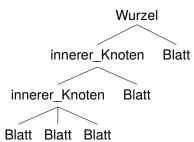
Binärbäume Suchbäume

Zusammen-

27. November 2018 P. Thiemann – Info I 6 / 40

Terminologie I

- Alle Knoten, denen keine Teilbäume zugeordnet sind, heißen Blätter.
- Knoten, die keine Blätter sind, heißen innere Knoten.



 Die Wurzel kann also ein Blatt sein (keine weiteren Teilbäume) oder ein innerer Knoten. Der Baum

Terminologie Beispiele

Binärbäume

Zucommo

Terminologie II



Wenn k_1 ein Knoten und k_2 die Wurzel eines zugeordneten Teilbaums ist, dann gilt:

Der Baum

Definition Terminologie

Beispiele

Binärbäume Suchbäume



- Wenn k₁ ein Knoten und k₂ die Wurzel eines zugeordneten Teilbaums ist, dann gilt:
 - \mathbf{k}_1 ist Elternknoten von k_2 ,

Der Baum

Definition Terminologie

Beispiele

Binärbäume Suchbäume

Zusammen-



A W

- Wenn k₁ ein Knoten und k₂ die Wurzel eines zugeordneten Teilbaums ist, dann gilt:
 - \blacksquare k_1 ist Elternknoten von k_2 ,
 - k_1 sowie der Elternknoten von k_1 sowie dessen Elternknoten usw. sind Vorgänger von k_2 .

Der Baum

Definition Terminologie

Beispiele

Binärbäume

Terminologie II





- Wenn k_1 ein Knoten und k_2 die Wurzel eines zugeordneten Teilbaums ist, dann gilt:
 - \mathbf{k}_1 ist Elternknoten von k_2 ,
 - k_1 sowie der Elternknoten von k_1 sowie dessen Elternknoten usw. sind Vorgänger von k_2 .
 - \blacksquare k_2 ist Kind von k_1 .

Der Baum

Terminologie

Beispiele

Binärbäume



- \blacksquare k_1 ist Elternknoten von k_2 ,
- k_1 sowie der Elternknoten von k_1 sowie dessen Elternknoten usw. sind Vorgänger von k_2 .
- \blacksquare k_2 ist Kind von k_1 .
- Alle Kinder von k_1 , deren Kinder, usw. sind Nachfolger von k_1 .

Der Baum

Terminologie Beispiele

Binärbäume

Terminologie II



SE SE

- Wenn k_1 ein Knoten und k_2 die Wurzel eines zugeordneten Teilbaums ist, dann gilt:
 - \blacksquare k_1 ist Elternknoten von k_2 ,
 - k_1 sowie der Elternknoten von k_1 sowie dessen Elternknoten usw. sind Vorgänger von k_2 .
 - \blacksquare k_2 ist Kind von k_1 .
 - Alle Kinder von k_1 , deren Kinder, usw. sind Nachfolger von k_1 .
- Bäume sind oft markiert. Die Markierung weist jedem Knoten eine Marke zu.

Der Baum

Terminologie Beispiele

Binärbäume

.

Zusammen

Terminologie II



A H

- Wenn k_1 ein Knoten und k_2 die Wurzel eines zugeordneten Teilbaums ist, dann gilt:
 - \blacksquare k_1 ist Elternknoten von k_2 ,
 - k_1 sowie der Elternknoten von k_1 sowie dessen Elternknoten usw. sind Vorgänger von k_2 .
 - \blacksquare k_2 ist Kind von k_1 .
 - Alle Kinder von k_1 , deren Kinder, usw. sind Nachfolger von k_1 .
- Bäume sind oft markiert. Die Markierung weist jedem Knoten eine Marke zu.
- Formal: Wenn K die Knotenmenge eines Baums ist und M eine Menge von Marken, dann ist die Markierung eine Abbildung $\mu: K \to M$.

Der Baum

Terminologie

Binärbäume

Suchbäume

Zusammen

Beispiel: Verzeichnisbaum



In Linux (und anderen Betriebssystemen) ist die Verzeichnisstruktur im Wesentlichen baumartig.

Der Baum

Definition Terminologie

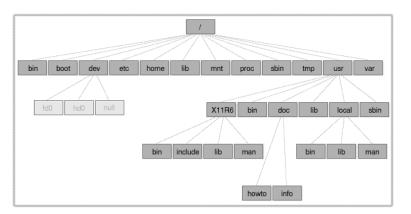
Beispiele

Binärbäume

Suchbäume

Beispiel: Verzeichnisbaum

In Linux (und anderen Betriebssystemen) ist die Verzeichnisstruktur im Wesentlichen baumartig.



Der Baum

Terminolog

Beispiele

Binärbäume Suchhäume

Beispiel: Syntaxbaum





Wenn die Struktur einer Sprache mit Hilfe einer formalen Grammatiken spezifiziert ist, dann kann der Satzaufbau durch sogenannte Syntaxbäume beschrieben werden.

Der Baum

Definition Terminolog

Beispiele

Binärbäume Suchhäume

Beispiel: Syntaxbaum

II IIBURG

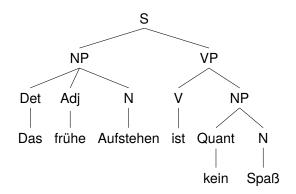
Wenn die Struktur einer Sprache mit Hilfe einer formalen Grammatiken spezifiziert ist, dann kann der Satzaufbau durch sogenannte Syntaxbäume beschrieben werden. Der Baum

Der Baum Definition

Beispiele

Dillarbaume

Zusammer







Bäume können arithmetische (und andere) Ausdrücke so darstellen, dass ihre Auswertung eindeutig (und einfach durchführbar) ist, ohne dass Klammern notwendig sind.

Der Baum

Terminolo

Beispiele

Binärbäume





Bäume können arithmetische (und andere) Ausdrücke so darstellen, dass ihre Auswertung eindeutig (und einfach durchführbar) ist, ohne dass Klammern notwendig sind.

■ Beispiel: (5+6) * 3 * 2

Der Baum

Terminolo Beispiele

Binärbäume

Diriarbaariic

Zusammen





Bäume können arithmetische (und andere) Ausdrücke so darstellen, dass ihre Auswertung eindeutig (und einfach durchführbar) ist, ohne dass Klammern notwendig sind.

■ Beispiel: (5+6) *3 * 2

■ Entspricht: ((5+6) * 3) * 2

Der Baum

Terminolog Beisniele

Binarbaume

Zusammer





Bäume können arithmetische (und andere) Ausdrücke so darstellen, dass ihre Auswertung eindeutig (und einfach durchführbar) ist, ohne dass Klammern notwendig sind.

■ Beispiel: (5+6) *3 * 2

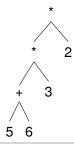
■ Entspricht: ((5+6) * 3) * 2

Operatoren als Markierung innerer Knoten, Zahlen als Markierung der Blätter: Der Baum Definition

Terminolog Beispiele

Binärbäume

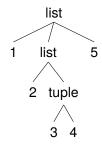
Odonbaam



Beispiel: Listen und Tupel als Bäume



- Jede Liste und jedes Tupel kann als Baum angesehen werden, bei dem der Typ die Knotenmarkierung ist und die Elemente die Teilbäume sind.
- Beispiel: [1, [2, (3, 4)], 5]



Der Baum Definition

Terminolo Beispiele

Discussion access



Der Baum

Binärbäume

Repräsentation Beispiel

Funktionen auf Bäumen

> Baumeigenschaften

Traversierung

Suchbäume

Zusammenfassung

Binärbäume



14 / 40

Der Baum

Binärbäume

- Der Binärbaum ist ein Spezialfall eines Baumes.
- Ein Binärbaum ist entweder leer oder besteht aus einem (Wurzel-) Knoten und zwei Teilbäumen.
- Für viele Anwendungsfälle angemessen.
- Funktionen über solchen Bäumen sind einfach definierbar.

Baumeigenschaf-

fassung



Der leere Baum wird durch None repräsentiert.

Der Baum

Binärbäume

Repräsentation

Beispiel Funktionen auf Bäumen

Baumeigenschaften

Traversierung

Suchbäume



- Der leere Baum wird durch None repräsentiert.
- Jeder andere Knoten wird durch ein Node-Objekt repräsentiert.

Der Baum

Repräsentation

Beisniel

Baumeigenschaf-

Traversierung

Suchhäume



FREIBL

- Der leere Baum wird durch None repräsentiert.
- Jeder andere Knoten wird durch ein Node-Objekt repräsentiert.
- Das Attribut mark enthält die Markierung.

Der Baum

Binärbäume

Repräsentation

Funktionen auf

Bäumen Baumeigenschaf-

Traversierung

C. . - b b =

Suchbäume





- Der leere Baum wird durch None repräsentiert.
- Jeder andere Knoten wird durch ein Node-Objekt repräsentiert.
- Das Attribut mark enthält die Markierung.
- Das Attribut left, enthält den linken Teilbaum.

Der Baum

Repräsentation

Baumeigenschaf-





- Der leere Baum wird durch None repräsentiert.
- Jeder andere Knoten wird durch ein Node-Objekt repräsentiert.
- Das Attribut mark enthält die Markierung.
- Das Attribut left enthält den linken Teilbaum.
- Das Attribut right enthält den rechten Teilbaum.

Der Baum

Binärbäume

Repräsentation

Funktionen au

Bäumen Baumeigenschaf-

Travereierung

O. . - l- l- = . . - - -

Suchbaum



FRE BC

- Der leere Baum wird durch None repräsentiert.
- Jeder andere Knoten wird durch ein Node-Objekt repräsentiert.
- Das Attribut mark enthält die Markierung.
- Das Attribut left enthält den linken Teilbaum.
- Das Attribut right enthält den rechten Teilbaum.
- Beispiele:

Der Baum

Binarbaum

Repräsentation

Funktionen auf

Baumeigenschaf-

Traversierung

Suchbäum

Zucommon

Zusammei fassung



FREE BU

- Der leere Baum wird durch None repräsentiert.
- Jeder andere Knoten wird durch ein Node-Objekt repräsentiert.
- Das Attribut mark enthält die Markierung.
- Das Attribut left enthält den linken Teilbaum.
- Das Attribut right enthält den rechten Teilbaum.
- Beispiele:
 - Der Baum bestehend aus dem einzigen Knoten mit der Markierung 8: Node (8, None, None)

Der Baum

Binarbaum

Repräsentation

Funktionen auf

Baumeigensc ten

Traversierung

Suchbäume

Zusammen



- Der leere Baum wird durch None repräsentiert.
- Jeder andere Knoten wird durch ein Node-Objekt repräsentiert.
- Das Attribut mark enthält die Markierung.
- Das Attribut left enthält den linken Teilbaum.
- Das Attribut right enthält den rechten Teilbaum.
- Beispiele:
 - Der Baum bestehend aus dem einzigen Knoten mit der Markierung 8: Node (8, None, None)
 - Der Baum mit Wurzel '+', linkem Teilbaum mit Blatt 5, rechtem Teilbaum mit Blatt 6:
 Node('+', Node(5, None, None), Node(6, None, None))

Der Baum

Binärbäum

Repräsentation

Funktionen auf

Baumeigenso ten

Traversierung

Suchbaume



```
Der Baum
```

Binärbäume

Repräsentation Beisniel

Funktionen auf Bäumen

Baumeigenschaften

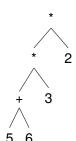
Traversierung

Suchbäume

```
class Node:
   def __init__(self, mark, left, right):
      self.mark = mark
      self.left = left
      self.right = right
```

Beispiel: Der Ausdrucksbaum





Der Baum

Rinärhäi

Ponrăcontation

Beispiel

Funktionen auf Bäumen Baumeigenschaf-

ten

Suchbäume

Sucribaurile

Zusammenfassung

wird folgendermaßen mit Node Objekten dargestellt:





Funktionsgerüst

```
def tree str (tree : Node) -> string:
    if tree is None:
        return "fill in"
    else:
        l_str = tree_str (tree.left)
        r_str = tree_str (tree.right)
        return "fill in"
```

Der Baum

Repräsentation Reisniel

Funktionen auf Bäumen

Baumeigenschaf-Traversierung

Suchhäume



Funktionsgerüst

```
def tree_str (tree : Node) -> string:
   if tree is None:
      return "filluin"

else:
      l_str = tree_str (tree.left)
      r_str = tree_str (tree.right)
      return "filluin"
```

Node Objekte enthalten selbst wieder Node Objekte (oder None) in den Attributen left und right. Der Baum

Binärbäu

Beispiel

Funktionen auf Bäumen Baumeigenschaf-

ten Traversierung

Suchbäume



Funktionsgerüst

```
def tree_str (tree : Node) -> string:
   if tree is None:
      return "fill_in"
   else:
      l_str = tree_str (tree.left)
      r_str = tree_str (tree.right)
      return "fill_in"
```

- Node Objekte enthalten selbst wieder Node Objekte (oder None) in den Attributen left und right.
- Zum Ausdrucken eines Node Objekts müssen auch die enhaltenen Node Objekte ausgedruckt werden.

Der Baum

Binärbäu

Beispiel

Bäumen Baumeigenschaf-

ten Traversierung

Suchbäume



F

Funktionsgerüst

```
def tree_str (tree : Node) -> string:
   if tree is None:
       return "fill_in"

else:
       l_str = tree_str (tree.left)
       r_str = tree_str (tree.right)
       return "fill_in"
```

- Node Objekte enthalten selbst wieder Node Objekte (oder None) in den Attributen left und right.
- Zum Ausdrucken eines Node Objekts müssen auch die enhaltenen Node Objekte ausgedruckt werden.
- Daher ist tree_str rekursiv, d.h. es wird in seiner eigenen Definition aufgerufen!

Der Baum

Binärbäur

Beispiel

Bäumen
Baumeigenschaf-

ten Traversierung

Suchbäume

Drucken von Bäumen erfolgt rekursiv



- Die rekursiven Aufrufe tree_str (tree.left) und tree_str (tree.left) erfolgen auf den Kindern des Knoten.
- Ergibt sich zwangsläufig aus der induktiven Definition!
- Rekursive Aufrufe auf den Teilbäumen sind Teil des Funktionsgerüsts, sobald eine baumartige Struktur bearbeitet werden soll.
- Die Alternative "tree is None" ergibt sich daraus, dass ein tree entweder None oder ein Node-Objekt ist.
- Alle Funktionen auf Binärbäumen verwenden dieses Gerüst.

Der Baum

Binärbäu

Repräsentation

Funktionen auf Bäumen

Baumeigenschaften Traversierung

Suchbäume

usammenssung

Drucken von Bäumen

Funktionsdefinition



20 / 40

```
def tree str (tree : Node) -> string:
    if tree is None:
        return "None"
    else:
        return ("Node("
            + repr(tree.mark) + ", "
            + tree_str (tree.left) + ", "
            + tree_str (tree.right) + ")")
```

Der Baum

Funktionen auf Bäumen

Baumeigenschaf-Traversierung

Suchhäume



■ Die Tiefe eines Knotens k (Abstand zur Wurzel) ist

Der Baum

Binärbäum

Depräsentation

Beispiel Funktionen auf

Baumeigenschaf-

ten Traversierung

Suchbäume

Suchbaum

Tiefe von Knoten, Höhe und Größe von (Binär-)Bäumen





- \blacksquare Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,

Der Baum

Binärbäume

Diriaibaaine

Beispiel Funktionen auf

Baumen Baumeigenschaf-

ten Traversierung

Suchhäume

Suchbaum



REBI

- Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,
 - \blacksquare *i* + 1, wenn *i* die Tiefe des Elternknotens ist.

Der Baum

Dinärhäum

Dinarbaume

Beispiel Funktionen au

Funktionen auf Bäumen Baumeigenschaf-

ten Traversierung

Cuchhäume

Suchbäume



- Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,
 - \blacksquare *i* + 1, wenn *i* die Tiefe des Elternknotens ist.
- Die Höhe eines Baumes ist die maximale Tiefe über alle Blätter:

Der Baum

Binärbäum

Denzioentation

Funktionen auf

Baumeigenschaf-

Travareiaruna

naverblerang

Suchbäume



REIBURG

- Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,
 - \blacksquare *i* + 1, wenn *i* die Tiefe des Elternknotens ist.
- Die Höhe eines Baumes ist die maximale Tiefe über alle Blätter:
 - -1 für den leeren Baum,

Der Baum

Binärbäume

Renräsentation

Funktionen auf

Baumeigenschaf-

Traversierung

Suchhäumo

Sucribaume

Zusammer fassung



Der Baum

- Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,
 - \blacksquare *i* + 1, wenn *i* die Tiefe des Elternknotens ist.
- Die Höhe eines Baumes ist die maximale Tiefe über alle Blätter:
 - -1 für den leeren Baum,
 - *m* + 1, wenn *m* die maximale Höhe aller der Wurzel zugeordneten Teilbäume ist.

Der Baum

Binärbäu

Repräsentation

Funktionen auf

Baumeigenschaften

Traversierung

Suchbäume



FRE

- Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,
 - \blacksquare *i* + 1, wenn *i* die Tiefe des Elternknotens ist.
- Die Höhe eines Baumes ist die maximale Tiefe über alle Blätter:
 - -1 für den leeren Baum,
 - m+1, wenn m die maximale H\u00f6he aller der Wurzel zugeordneten Teilb\u00e4ume ist.
- Die Größe eines Baumes ist die Anzahl seiner Knoten.

Der Baum

Binärbäur

Beispiel

Funktionen auf Bäumen Baumeigenschaf-

ten

naversierung

Sucribaume

Tiefe von Knoten, Höhe und Größe von (Binär-)Bäumen



7. H

- Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,
 - \blacksquare *i* + 1, wenn *i* die Tiefe des Elternknotens ist.
- Die Höhe eines Baumes ist die maximale Tiefe über alle Blätter:
 - -1 für den leeren Baum,
 - *m* + 1, wenn *m* die maximale Höhe aller der Wurzel zugeordneten Teilbäume ist.
- Die Größe eines Baumes ist die Anzahl seiner Knoten.
 - 0 für den leeren Baum,

Der Baum

Binärbäun

Repräsentation Beispiel

Funktionen auf Bäumen

Baumeigenschaften

Traversierung

Suchbaume

Tiefe von Knoten, Höhe und Größe von (Binär-)Bäumen



Der Baum

- Die Tiefe eines Knotens k (Abstand zur Wurzel) ist
 - 0, falls k die Wurzel ist,
 - \blacksquare *i* + 1, wenn *i* die Tiefe des Elternknotens ist.
- Die Höhe eines Baumes ist die maximale Tiefe über alle Blätter:
 - -1 für den leeren Baum,
 - m + 1, wenn m die maximale H\u00f6he aller der Wurzel zugeordneten Teilb\u00e4ume ist.
- Die Größe eines Baumes ist die Anzahl seiner Knoten.
 - 0 für den leeren Baum,
 - s+1, wenn s die Summe der Größen der Teilbäume ist.

Der Baum

Binärbäur

Beispiel

Funktionen auf Bäumen Baumeigenschaf-

ten

_ ...

Ouchbaume

Induktive Definition von Höhe und Größe von Binärbäumen



 $height(tree) = \begin{cases} -1, \\ 1 + \max(& height(tree.left), \\ & height(tree.right)), \end{cases}$ if tree is empty otherwise.

Der Baum

Baumeigenschaf-



Der Baum

Binärbäun

Penräcental

Funktionen auf

Bäumen

Baumeigenschaf-

ten

Traversierung

Suchbäume

7

$$height(tree) = \begin{cases} -1, & \text{if } tree \text{ is empty} \\ 1 + \max(& height(tree.left), \\ & height(tree.right)), & \text{otherwise.} \end{cases}$$

$$size(tree) = \begin{cases} 0, & \text{if } tree \text{ is empty;} \\ 1 & +size(tree.left) \\ & +size(tree.right)), & \text{otherwise.} \end{cases}$$

Funktionen für Höhe und Größe



Höhe und Größe von Binärbäumen

```
def height(tree):
    if (tree is None):
        return -1
    else:
        return (max (height (tree.left),
                    height(tree.right)) + 1)
def size(tree):
    if (tree is None):
        return 0
    else:
        return(size(tree.left)
             + size(tree.right) + 1)
tree = Node('*', Node('+', Node(6, None, None), Node(5,
                  Node(1, None, None))
```

Der Baum

Baumeigenschaf-Traversierung

Suchhäume



Oft sollen alle Knoten eines Baumes besucht und bearbeitet werden.

Der Baum

Binärbäume

Daniel

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

Traversierung

Suchbäume

Sucribaum

Der Baum

Reisniel Funktionen auf

Baumeigenschaf-

Traversierung

Suchhäume

- Oft sollen alle Knoten eines Baumes besucht und bearbeitet werden.
- 3 Vorgehensweisen (Traversierungen) sind üblich:



NE TO THE TOTAL

- Oft sollen alle Knoten eines Baumes besucht und bearbeitet werden.
- 3 Vorgehensweisen (Traversierungen) sind üblich:
 - Pre-Order (Hauptreihenfolge): Zuerst der Knoten selbst, dann der linke, danach der rechte Teilbaum

Der Baum

Binärbäum

Repräsentation

Funktionen auf Bäumen

Baumeigenschaften

Traversierung

Suchbäume

Zusammen

24 / 40



PRE I

- Oft sollen alle Knoten eines Baumes besucht und bearbeitet werden.
- 3 Vorgehensweisen (Traversierungen) sind üblich:
 - Pre-Order (Hauptreihenfolge): Zuerst der Knoten selbst, dann der linke, danach der rechte Teilbaum
 - Post-Order (Nebenreihenfolge): Zuerst der linke, danach der rechte Teilbaum, zum Schluss der Knoten selbst

Der Baum

Binärbäur

Repräsentation

Funktionen auf Bäumen

Baumeigenschaften Traversierung

Suchhäum

Suchbäume



- Oft sollen alle Knoten eines Baumes besucht und bearbeitet werden.
- 3 Vorgehensweisen (Traversierungen) sind üblich:
 - Pre-Order (Hauptreihenfolge): Zuerst der Knoten selbst, dann der linke, danach der rechte Teilbaum
 - Post-Order (Nebenreihenfolge): Zuerst der linke, danach der rechte Teilbaum, zum Schluss der Knoten selbst
 - In-Order (symmetrische Reihenfolge): Zuerst der linke Teilbaum, dann der Knoten selbst, danach der rechte Teilbaum

Der Baum

Binärbäum

Repräsentation

Funktionen auf

Baumeigenschaften

Traversierung

Suchbäume

Zusammen-



- Oft sollen alle Knoten eines Baumes besucht und bearbeitet werden.
- 3 Vorgehensweisen (Traversierungen) sind üblich:
 - Pre-Order (Hauptreihenfolge): Zuerst der Knoten selbst, dann der linke, danach der rechte Teilbaum
 - Post-Order (Nebenreihenfolge): Zuerst der linke, danach der rechte Teilbaum, zum Schluss der Knoten selbst
 - In-Order (symmetrische Reihenfolge): Zuerst der linke Teilbaum, dann der Knoten selbst, danach der rechte Teilbaum
- Manchmal auch Reverse In-Order (anti-symmetrische Reihenfolge): Rechter Teilbaum, Knoten, dann linker Teilbaum

Der Baum

Binärbäum

Beispiel Funktionen auf

ten Traversierung

Suchhäume

Sucribaurrie

zusammei assung



AR NE E

- Oft sollen alle Knoten eines Baumes besucht und bearbeitet werden.
- 3 Vorgehensweisen (Traversierungen) sind üblich:
 - Pre-Order (Hauptreihenfolge): Zuerst der Knoten selbst, dann der linke, danach der rechte Teilbaum
 - Post-Order (Nebenreihenfolge): Zuerst der linke, danach der rechte Teilbaum, zum Schluss der Knoten selbst
 - In-Order (symmetrische Reihenfolge): Zuerst der linke Teilbaum, dann der Knoten selbst, danach der rechte Teilbaum
- Manchmal auch Reverse In-Order (anti-symmetrische Reihenfolge): Rechter Teilbaum, Knoten, dann linker Teilbaum
- Auch das Besuchen nach Tiefenlevel von links nach rechts (level-order) ist denkbar

Der Baum

Binärbäur

Repräsentation Beispiel

Funktionen auf Bäumen

Traversierung

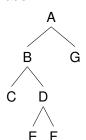
Suchbäume

Zusammen-

Pre-Order Ausgabe eines Baums



■ Gebe Baum *pre-order* aus



Der Baum

Binärbäume

Beispiel Funktionen auf Bäumen

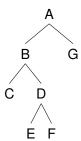
Baumeigenschaf-

Traversierung

Suchbäume



■ Gebe Baum pre-order aus



Ausgabe: A

Der Baum

Binärbäume

Dillalbauli

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

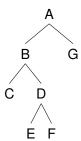
Traversierung

Suchbäume

Ouchbaum



■ Gebe Baum pre-order aus



Ausgabe: A

Der Baum

Binärbäume

Dillalbauli

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

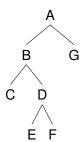
Traversierung

Suchbäume

Ouchbaum



■ Gebe Baum pre-order aus



Ausgabe: A B

Der Baum

Binärbäume

Diriarbaui

Beispiel Funktionen auf

Bäumen Baumeigenschaf-

ten

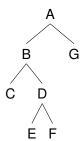
Traversierung

Suchbäume

Guombaam



■ Gebe Baum pre-order aus



Ausgabe: A B C

Der Baum

Binärbäume

Diriaibaai

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

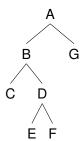
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum *pre-order* aus



Ausgabe: A B C D

Der Baum

Binärbäume

Diriarbaan

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

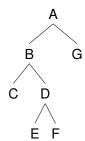
Traversierung

Suchhäume

Guonbaam



■ Gebe Baum pre-order aus



Ausgabe: A B C D E

Der Baum

Binärbäume

Diriaibaai

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

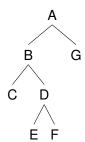
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum pre-order aus



Ausgabe: A B C D E F

Der Baum

Binärbäume

Beispiel Funktionen auf

Baumeigenschaf-

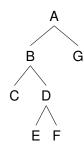
Traversierung

Suchbäume

Suchbaum



■ Gebe Baum pre-order aus



Ausgabe: A B C D E F G

Der Baum

Binärbäume

Diriaibaai

Beispiel Funktionen auf

Baumeigenschaf-

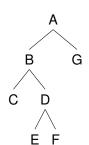
Traversierung

Suchbäume

Suchbaum



■ Gebe Baum *post-order* aus



Der Baum

Binärbäume

Repräsentation Beisniel

Funktionen auf Bäumen

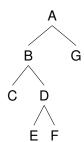
Baumeigenschaf-

Traversierung

Suchbäume



■ Gebe Baum post-order aus



Ausgabe: C

Der Baum

Binärbäume

Repräsentatio Beispiel

Funktionen auf Bäumen

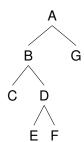
Baumeigenschaf-

Traversierung

Suchbäume



■ Gebe Baum post-order aus



Ausgabe: C

Der Baum

Binärbäume

Repräsentatio Beispiel

Funktionen auf Bäumen

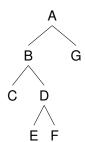
Baumeigenschaf-

Traversierung

Suchbäume



■ Gebe Baum post-order aus



Ausgabe: C E

Der Baum

Binärbäume

Diriaibaai

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

Traversierung

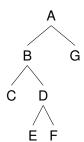
Suchbäume

Zusammen-

fassung



■ Gebe Baum post-order aus



■ Ausgabe: C E F

Der Baum

Binärbäume

Repräsentation Beispiel

Funktionen auf

Baumeigenschaf-

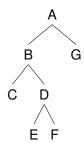
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum *post-order* aus



■ Ausgabe: C E F D

Der Baum

Binärbäume

Diriarbaar

Beispiel Funktionen auf

Baumeigenschaf-

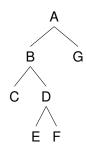
Traversierung

Suchhäume

Ouchbaum



■ Gebe Baum post-order aus



■ Ausgabe: C E F D B

Der Baum

Binärbäume

Repräsentation Beispiel

Funktionen auf

Baumeigenschaf-

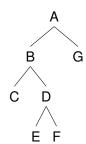
Traversierung

Suchbäume

Suchbaum



■ Gebe Baum post-order aus



■ Ausgabe: C E F D B G

Der Baum

Binärbäume

Repräsentation Beispiel

Funktionen auf

Baumeigenschaf-

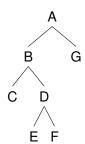
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum post-order aus



■ Ausgabe: C E F D B G A

Der Baum

Binärbäume

Diriaibau

Beispiel Funktionen a

Baumeigenschaf-

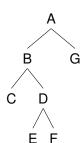
Traversierung

Suchhäume

Ouchbaum



■ Gebe Baum in-order aus.



Der Baum

Binärbäume

Repräsentation

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

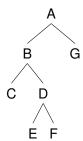
Traversierung

Suchbäume

Odonbaann



■ Gebe Baum in-order aus.



Ausgabe: C

Der Baum

Binärbäume

Diriaibaui

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

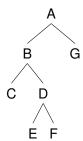
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum in-order aus.



Ausgabe: C

Der Baum

Binärbäume

Diriaibaui

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

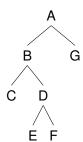
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum in-order aus.



■ Ausgabe: C B

Der Baum

Binärbäume

Dillalbauli

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

Traversierung

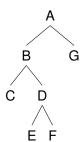
Suchbäume

Zusammen-

fassung



■ Gebe Baum in-order aus.



■ Ausgabe: C B E

Der Baum

Binärbäume

Dillaibauli

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

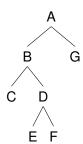
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum in-order aus.



■ Ausgabe: C B E D

Der Baum

Binärbäume

Dinarbaui

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

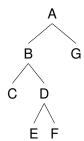
Traversierung

Suchbäume

Suchbaum



■ Gebe Baum in-order aus.



■ Ausgabe: C B E D F

Der Baum

Binärbäume

Diriarbaar

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

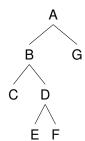
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum in-order aus.



■ Ausgabe: C B E D F A

Der Baum

Binärbäume

- Dinarbaun

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

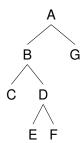
Traversierung

Suchhäume

Suchbaum



■ Gebe Baum in-order aus.



■ Ausgabe: C B E D F A G

Der Baum

Binärbäume

Diriarbaar

Beispiel Funktionen auf Bäumen

Baumeigenschaf-

Traversierung

Suchhäume

Suchbäum



<u>___</u>__

Post-Order Printing

Der Baum

Binärbäume

Benräsentation

Beispiel Funktionen au

Bäumen Baumeigenschaf-

Traversierung

.....

Suchbäume

Die *post-order* Ausgabe eines arithmetischen Ausdrucks heißt auch <u>umgekehrt polnische</u> oder <u>Postfix-Notation</u> (HP-Taschenrechner, Programmiersprachen *Forth* und *PostScript*) Der Baum

Binärbäum

Repräsentation Beisniel

Funktionen auf

Baumen Baumeigenschaf-

Traversierung

Suchbäume

Odonbaame



Der Baum

Binärbäume

Suchbäume

Definition Suche Aufbau



Z W Z

■ Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.

Der Baum

Binärbäume

Suchbäume Definition

Suche





- Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.
- Ein Suchbaum ist ein binärer Baum, der die Suchbaumeigenschaften erfüllt:

Der Baum

Binärbäume

Suchhäume

Definition Suche

Aufbau

Zusammen-

fassung



- Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.
- Ein Suchbaum ist ein binärer Baum, der die Suchbaumeigenschaften erfüllt:
 - Alle Markierungen im linken Teilbaum sind *kleiner* als die aktuelle Knotenmarkierung, alle Markierungen im rechten Teilbaum sind *größer*.

Der Baum

Binärbäume

Suchbäum

Definition Suche

Aufbau



<u>r</u>

- Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.
- Ein Suchbaum ist ein binärer Baum, der die Suchbaumeigenschaften erfüllt:
 - Alle Markierungen im linken Teilbaum sind kleiner als die aktuelle Knotenmarkierung, alle Markierungen im rechten Teilbaum sind größer.
- Suchen nach einem Item *m*: Vergleiche mit Markierung im aktuellem Knoten,

Der Baum

Binarbaume

Definition

Aufbau



- Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.
- Ein Suchbaum ist ein binärer Baum, der die Suchbaumeigenschaften erfüllt:
 - Alle Markierungen im linken Teilbaum sind kleiner als die aktuelle Knotenmarkierung, alle Markierungen im rechten Teilbaum sind größer.
- Suchen nach einem Item *m*: Vergleiche mit Markierung im aktuellem Knoten,
 - wenn gleich, stoppe und gebe True zurück,

Der Baum

Binärbäume

Definition

Aufbau



azu,

31/40

- Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.
- Ein Suchbaum ist ein binärer Baum, der die Suchbaumeigenschaften erfüllt:
 - Alle Markierungen im linken Teilbaum sind kleiner als die aktuelle Knotenmarkierung, alle Markierungen im rechten Teilbaum sind größer.
- Suchen nach einem Item *m*: Vergleiche mit Markierung im aktuellem Knoten,
 - wenn gleich, stoppe und gebe True zurück,
 - wenn *m* kleiner ist, suche im linken Teilbaum,

Der Baum

Binärbäume

Definition

Suche



- Der Baum
 - Binärbäume
 - Suchbäum
 - Suche

- Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.
- Ein Suchbaum ist ein binärer Baum, der die Suchbaumeigenschaften erfüllt:
 - Alle Markierungen im linken Teilbaum sind kleiner als die aktuelle Knotenmarkierung, alle Markierungen im rechten Teilbaum sind größer.
- Suchen nach einem Item *m*: Vergleiche mit Markierung im aktuellem Knoten,
 - wenn gleich, stoppe und gebe True zurück,
 - wenn *m* kleiner ist, suche im linken Teilbaum,
 - wenn *m* größer ist, such im rechten Teilbaum.

Suchbäume



- Der Baum
- Binärbäume
- Definition Suche
- Aufbau

- Suchbäume realisieren Wörterbücher und dienen dazu, Items schnell wieder zu finden.
- Ein Suchbaum ist ein binärer Baum, der die Suchbaumeigenschaften erfüllt:
 - Alle Markierungen im linken Teilbaum sind kleiner als die aktuelle Knotenmarkierung, alle Markierungen im rechten Teilbaum sind größer.
- Suchen nach einem Item *m*: Vergleiche mit Markierung im aktuellem Knoten,
 - wenn gleich, stoppe und gebe True zurück,
 - wenn *m* kleiner ist, suche im linken Teilbaum,
 - wenn *m* größer ist, such im rechten Teilbaum.
- Suchzeit ist proportional zur Höhe des Baums! Im besten Fall logarithmisch in der Größe des Baums.



꽃

Search in search tree

```
Der Baum
def search(tree, item):
                                                      Binärbäume
    if tree is None:
        return False
    elif tree.mark == item:
                                                       Suche
                                                       Aufbau
        return True
    elif tree.mark > item:
                                                      fassung
        return search(tree.left, item)
    else:
        return search(tree.right, item)
# smaller values left, bigger values in right subtree
nums = Node(10, Node(5, leaf(1), None),
                 Node (15, leaf (12), leaf (20))
print(search(nums, 12))
```

Immutable — unveränderlich



■ Aufruf insert(tree, item) für das Einsortieren von item in tree

Der Baum

Binärbäume

Suchbäume Definition

Aufbau

Immutable — unveränderlich





- Aufruf insert(tree, item) für das Einsortieren von item in tree
- Ist tree leer, so wird der Knoten leaf(item) zurückgegeben.

Der Baum

Binärbäume Suchbäume

Definition

Aufbau

Immutable — unveränderlich



2

- Aufruf insert(tree, item) für das Einsortieren von item in tree
- Ist tree leer, so wird der Knoten leaf(item) zurückgegeben.
- Wenn die Markierung tree.mark größer als item ist, wird item in den linken Teilbaum eingesetzt und der Baum rekonstruiert (das erhält die Suchbaumeigenschaft!).

Der Baum

Binärbäume

Definition

Aufbau

Immutable — unveränderlich



- Aufruf insert(tree, item) für das Einsortieren von item in tree
- Ist tree leer, so wird der Knoten leaf(item) zurückgegeben.
- Wenn die Markierung tree.mark größer als item ist, wird item in den linken Teilbaum eingesetzt und der Baum rekonstruiert (das erhält die Suchbaumeigenschaft!).
- Falls tree.mark kleiner als item ist, entsprechend.

Der Baum

Binärbäume

Definition

Aufbau



33 / 40

2

- Aufruf insert(tree, item) für das Einsortieren von item in tree
- Ist tree leer, so wird der Knoten leaf(item) zurückgegeben.
- Wenn die Markierung tree.mark größer als item ist, wird item in den linken Teilbaum eingesetzt und der Baum rekonstruiert (das erhält die Suchbaumeigenschaft!).
- Falls tree.mark kleiner als item ist, entsprechend.
- Falls tree.mark == item müssen wir nichts machen.

Der Baum

Binärbäume

Definition

Aufbau

Immutable — unveränderlich

Creating a search tree

```
def insert(tree, item):
    if tree is None:
        return leaf(item)
    elif tree.mark > item:
        return Node (tree.mark,
                      insert (tree.left, item),
                      tree.right)
    elif tree mark < item:
        return Node (tree.mark,
                      tree.left,
                      insert(tree.right, item))
    else:
        return tree
```

Der Baum

Aufbau



```
Der Baum
```

Binärbäume

Suchhäume

Suche

```
def insertall (tree, lst):
    for key in lst
        tree = insert (tree, key)
    return tree

bst = insertall (None, [10, 15, 20, 12, 5, 1])
```

Mutable — veränderlich



Creating a mutable search tree

```
def insertm(tree, item):
    if tree is None:
        return leaf(item)
    if tree.mark > item:
        tree.left = insertm(tree.left, item)
    elif tree.mark < item:
        tree.right = insertm(tree.right, item)
    return tree</pre>
```

Der Baum

Binärbäum

Suchbäume

Suche

Suchbaumaufbau

Mutable — verönderlich



```
Der Baum
```

Binärbäume

Suchbäume

Suche

```
def insertmall (tree, lst):
    for key in lst
        tree = insertm (tree, key)
    return tree
bst = insertmall (None, [10, 15, 20, 12, 5, 1])
```



Der Baum

Binärbäume

Suchbäume

Zusammenfassung

Zusammenfassung

27. November 2018 P. Thiemann – Info I 38 / 40

Der Baum ist eine Struktur, die in der Informatik allgegenwärtig ist.

Der Baum

Binärbäume

Suchbäume

- Der Baum ist eine Struktur, die in der Informatik allgegenwärtig ist.
- Binärbäume sind Bäume, bei denen jeder Knoten genau zwei Teilbäume besitzt.



Ž

- Der Baum ist eine Struktur, die in der Informatik allgegenwärtig ist.
- Binärbäume sind Bäume, bei denen jeder Knoten genau zwei Teilbäume besitzt.
- Operationen über (Binär-)Bäumen lassen sich einfach als rekursive Funktionen implementieren.

Der Baum

Suchhäume

Zucommon



Der Baum

Binärbäume

Suchbäume

- Der Baum ist eine Struktur, die in der Informatik allgegenwärtig ist.
- Binärbäume sind Bäume, bei denen jeder Knoten genau zwei Teilbäume besitzt.
- Operationen über (Binär-)Bäumen lassen sich einfach als rekursive Funktionen implementieren.
- Es gibt drei Hauptarten der Traversierung von Binärbäumen.



- Der Baum ist eine Struktur, die in der Informatik allgegenwärtig ist.
- Binärbäume sind Bäume, bei denen jeder Knoten genau zwei Teilbäume besitzt.
- Operationen über (Binär-)Bäumen lassen sich einfach als rekursive Funktionen implementieren.
- Es gibt drei Hauptarten der Traversierung von Binärbäumen.
- Suchbäume sind Binärbäume, die die Suchbaumeigenschaft besitzen, d.h. in linken Teilbaum sind nur kleinere, im rechten nur größere Markierungen.

Der Baum

Dillarbaume



- Der Baum
 - Binärbäume
- Zusammenfassung

- Der Baum ist eine Struktur, die in der Informatik allgegenwärtig ist.
- Binärbäume sind Bäume, bei denen jeder Knoten genau zwei Teilbäume besitzt.
- Operationen über (Binär-)Bäumen lassen sich einfach als rekursive Funktionen implementieren.
- Es gibt drei Hauptarten der Traversierung von Binärbäumen.
- Suchbäume sind Binärbäume, die die Suchbaumeigenschaft besitzen, d.h. in linken Teilbaum sind nur kleinere, im rechten nur größere Markierungen.
- Das Suchen und Einfügen kann durch einfache rekursive Funktionen realisiert werden. Sortierte Ausgabe ist auch sehr einfach!