

Datenstrukturen (Kapitel 3)

Prüfungsvorbereitung

Aufgabe 1

Nenne drei unterschiedliche Informatik-Anwendungen, in denen Bäume als Datenstruktur vorkommen.

Aufgabe 1

- ▶ HTML-Seiten
- ▶ hierarchische Dateisysteme
- ▶ Parse-Trees

Aufgabe 2

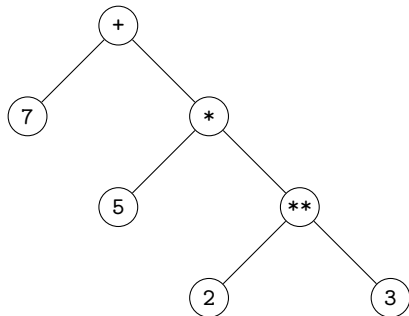
Stelle den Python-Ausdruck

`7 + 5 * 2 ** 3`

in Form eines Syntaxbaums dar. Beachte dabei die Standardprioritäten der Operatoren.

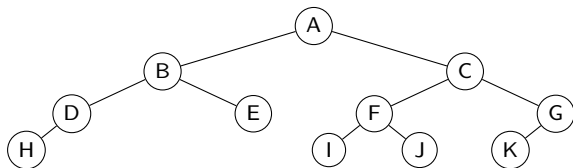
Aufgabe 2

7 + 5 * 2 ** 3



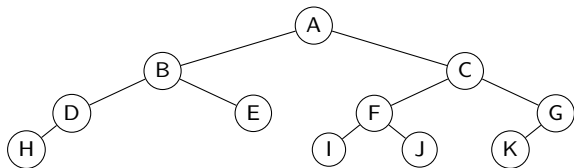
Aufgabe 3

Beantworte die Fragen zum folgenden Baum mit der Wurzel A:



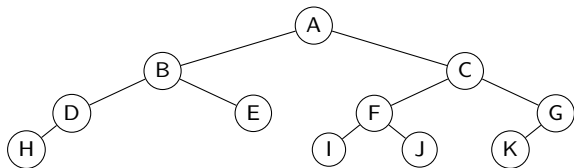
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C?
- (b) Welche Geschwister hat der Knoten mit dem Schlüssel H?
- (c) Welchen Elternknoten hat der Knoten mit dem Schlüssel E?
- (d) Welches sind die Blätter des Baums?
- (e) Welches sind die inneren Knoten des Baums?
- (f) Welche Tiefe hat der Knoten mit dem Schlüssel E?
- (g) Welche Höhe hat der Knoten mit dem Schlüssel J?
- (h) Welche Höhe hat der Baum?

Aufgabe 3



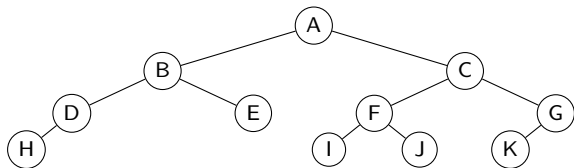
(a) Welche Kinder hat der Knoten mit dem Schlüssel C?

Aufgabe 3



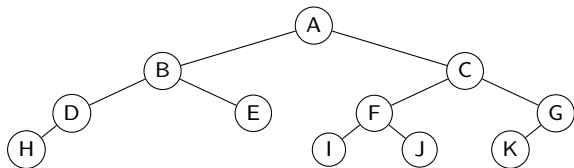
(a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**

Aufgabe 3



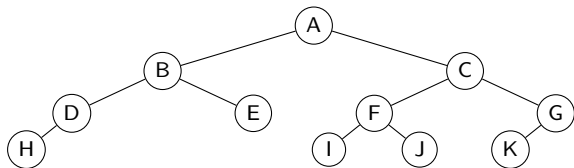
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H?

Aufgabe 3



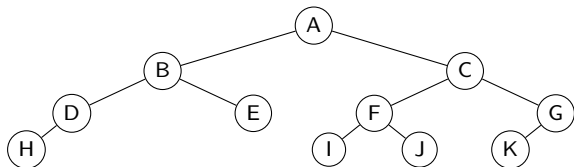
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**

Aufgabe 3



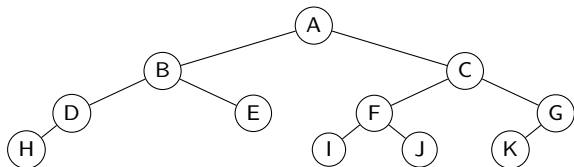
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E?

Aufgabe 3



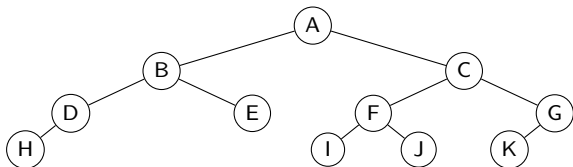
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**

Aufgabe 3



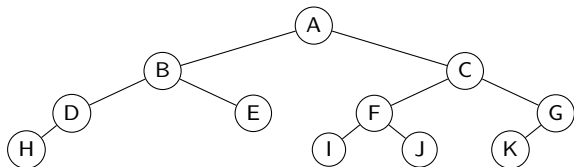
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums?

Aufgabe 3



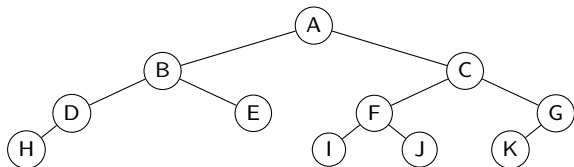
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**

Aufgabe 3



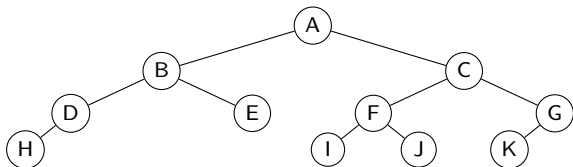
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums?

Aufgabe 3



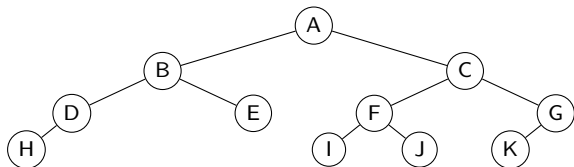
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums? **A, B, C, D, F, G**

Aufgabe 3



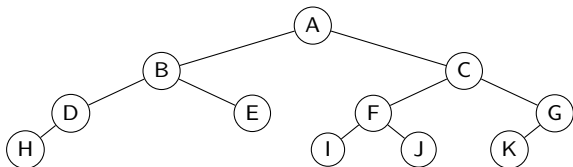
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums? **A, B, C, D, F, G**
- (f) Welche Tiefe hat der Knoten mit dem Schlüssel E?

Aufgabe 3



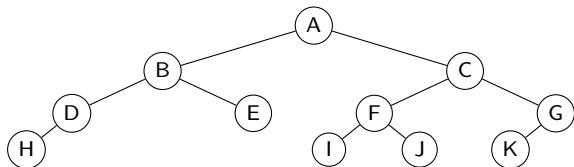
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums? **A, B, C, D, F, G**
- (f) Welche Tiefe hat der Knoten mit dem Schlüssel E? **2**

Aufgabe 3



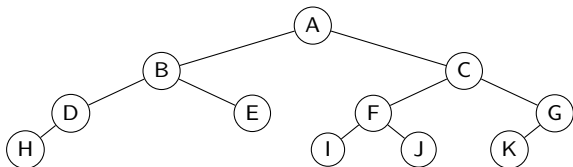
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums? **A, B, C, D, F, G**
- (f) Welche Tiefe hat der Knoten mit dem Schlüssel E? **2**
- (g) Welche Höhe hat der Knoten mit dem Schlüssel J?

Aufgabe 3



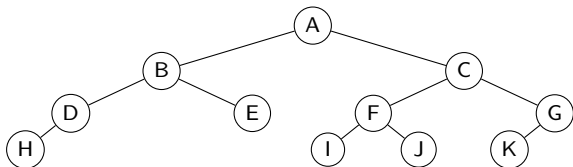
- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums? **A, B, C, D, F, G**
- (f) Welche Tiefe hat der Knoten mit dem Schlüssel E? **2**
- (g) Welche Höhe hat der Knoten mit dem Schlüssel J? **0**

Aufgabe 3



- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums? **A, B, C, D, F, G**
- (f) Welche Tiefe hat der Knoten mit dem Schlüssel E? **2**
- (g) Welche Höhe hat der Knoten mit dem Schlüssel J? **0**
- (h) Welche Höhe hat der Baum?

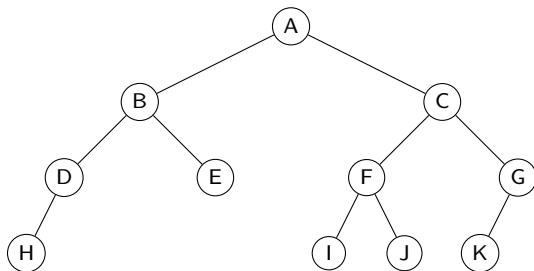
Aufgabe 3



- (a) Welche Kinder hat der Knoten mit dem Schlüssel C? **F, G**
- (b) Welche Geschwister hat der Knoten H? **keine**
- (c) Welche Eltern hat der Knoten mit dem Schlüssel E? **B**
- (d) Welches sind die Blätter des Baums? **H, E, I, J, K**
- (e) Welches sind die inneren Knoten des Baums? **A, B, C, D, F, G**
- (f) Welche Tiefe hat der Knoten mit dem Schlüssel E? **2**
- (g) Welche Höhe hat der Knoten mit dem Schlüssel J? **0**
- (h) Welche Höhe hat der Baum? **3**

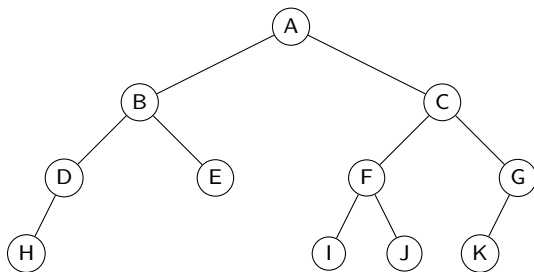
Aufgabe 4

Beschreibe den Typ des Baums mit dem richtigen Fachausdruck.



Aufgabe 4

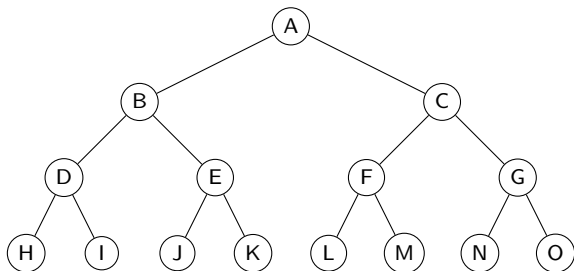
balancierter Binärbaum



Die Tiefe der Blätter unterscheiden sich höchstens um 1.

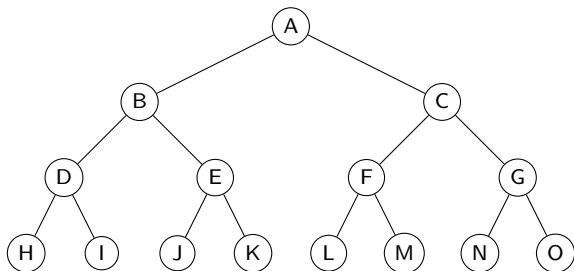
Aufgabe 5

Beschreibe den Typ des Baums mit dem richtigen Fachausdruck.



Aufgabe 5

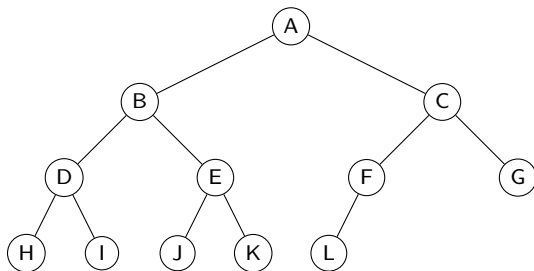
perfekter Binärbaum



Ein voller Binärbaum, dessen Blätter alle dieselbe Tiefe haben.

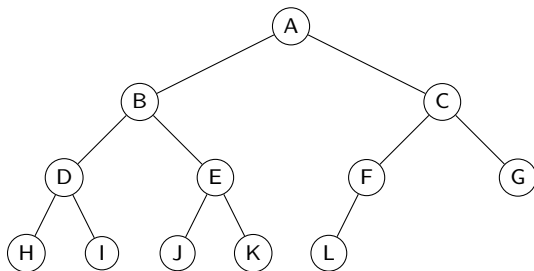
Aufgabe 6

Beschreibe den Typ des Baums mit dem richtigen Fachausdruck.



Aufgabe 6

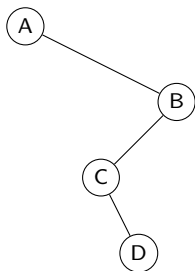
vollständiger Binärbaum



Ein balancierter Binärbaum, bei dem die tieferen Blätter alle links stehen.

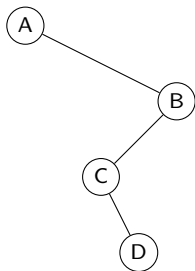
Aufgabe 7

Beschreibe den Typ des Baums mit dem richtigen Fachausdruck.



Aufgabe 7

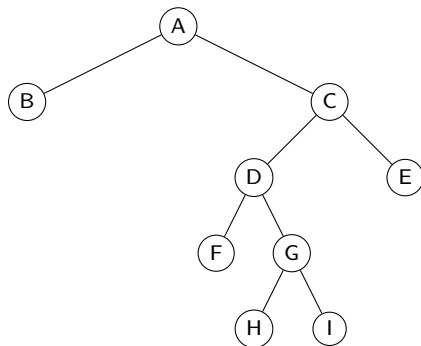
entarteteter Baum



Jeder Knoten hat höchstens ein Kind.

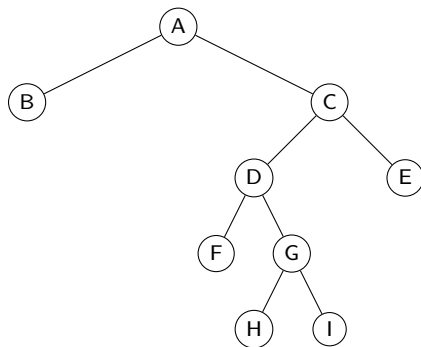
Aufgabe 8

Beschreibe den Typ des Baums mit dem richtigen Fachausdruck.



Aufgabe 8

voller Binärbaum



Alle inneren Knoten haben zwei Kinder.

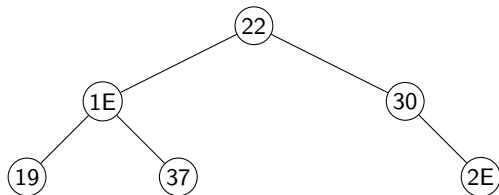
Aufgabe 9

Das folgende Speicherabbild enthält einen Baum. Die Adresse 0x25 enthält einen Zeiger auf den Wurzelknoten. Jeder Knoten besteht aus drei benachbarten Speicherzellen, wobei der erste Knoten den Schlüssel, der zweite Knoten die Referenz auf einen allfälligen linken Teilbaum und der dritte Knoten die Referenz auf einen allfälligen rechten Teilbaum enthält. 0x00 bezeichnet den NULL-Zeiger. Skizziere diesen Baum mit seinen Schlüssel.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	⊗	00	0B	41	1E	2F	10	00	00	00	00	00	00	00	00	22
1	37	00	00	00	3A	0F	00	32	41	00	00	00	00	00	00	00
2	00	00	00	00	00	32	00	00	00	0C	14	17	00	00	46	19
3	00	00	22	04	3A	00	00	32	00	00	30	00	4C	24	46	00
4	00	00	00	00	00	00	00	02	04	00	00	00	2E	00	00	00

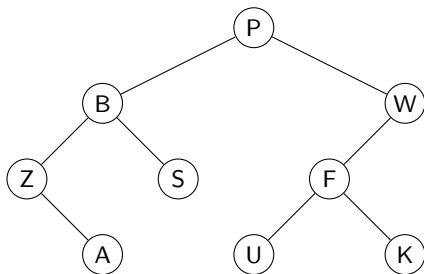
Aufgabe 9

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	⊗	00	0B	41	1E	2F	10	00	00	00	00	00	00	00	00	22
1	37	00	00	00	3A	0F	00	32	41	00	00	00	00	00	00	00
2	00	00	00	00	00	32	00	00	00	0C	14	17	00	00	46	19
3	00	00	22	04	3A	00	00	32	00	00	80	00	4C	24	46	00
4	00	00	00	00	00	00	00	02	04	00	00	00	2E	00	00	00



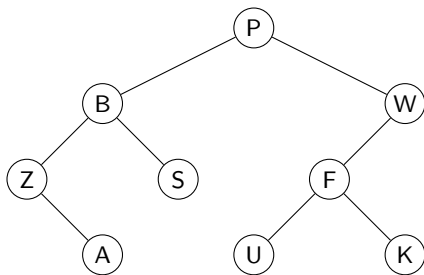
Aufgabe 10

Gib die Schlüsselreihenfolge bei den folgenden Traversierungen an.



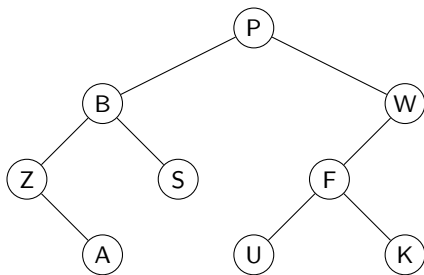
- (a) Preorder
- (b) Inorder
- (c) Postorder

Aufgabe 10



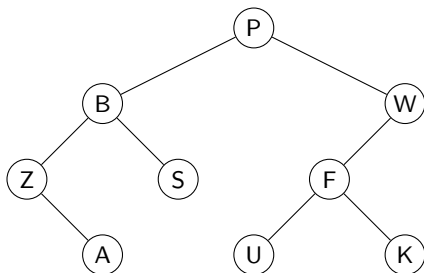
(a) Preorder:

Aufgabe 10



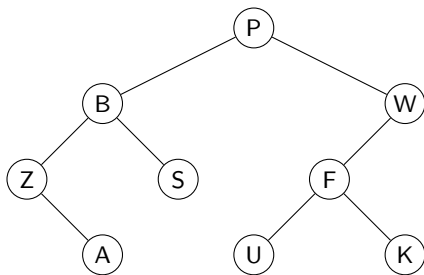
(a) Preorder: P

Aufgabe 10



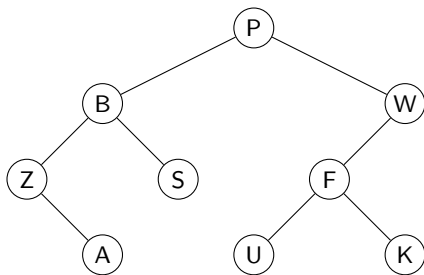
(a) Preorder: P B

Aufgabe 10



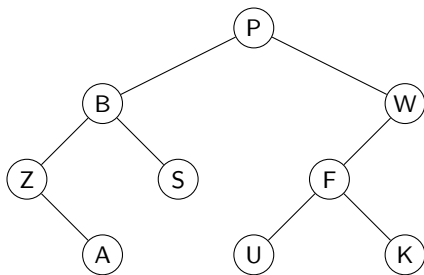
(a) Preorder: P B Z

Aufgabe 10



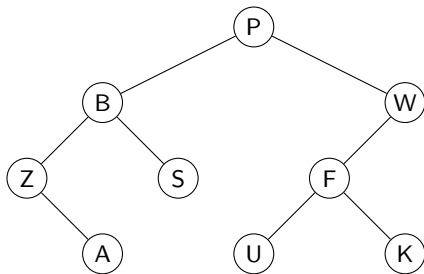
(a) Preorder: P B Z A

Aufgabe 10



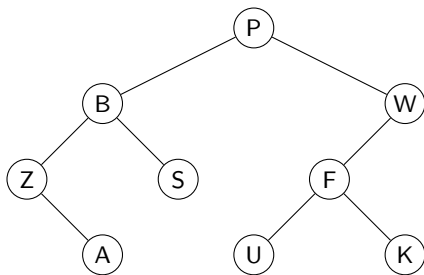
(a) Preorder: P B Z A S

Aufgabe 10



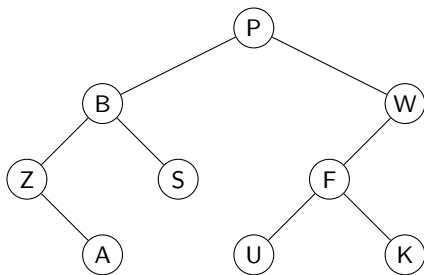
(a) Preorder: P B Z A S W

Aufgabe 10



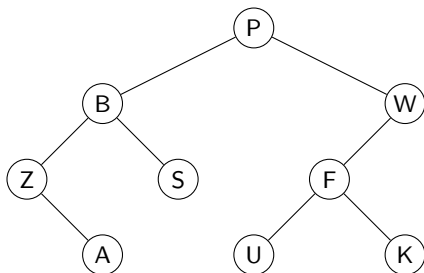
(a) Preorder: P B Z A S W F

Aufgabe 10



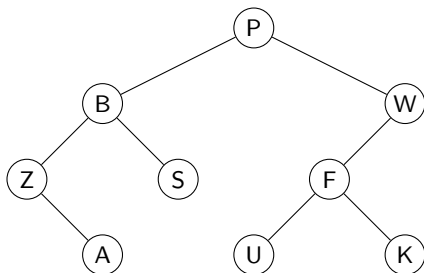
(a) Preorder: P B Z A S W F U

Aufgabe 10



(a) Preorder: P B Z A S W F U K

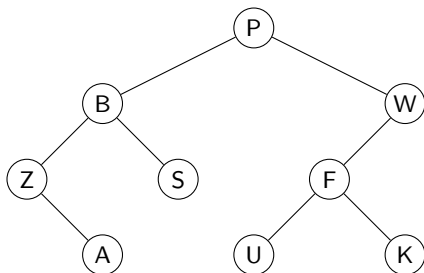
Aufgabe 10



(a) Preorder: P B Z A S W F U K

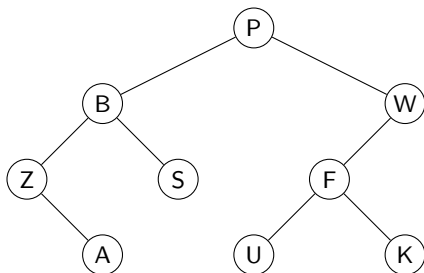
(b) Inorder:

Aufgabe 10



- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z

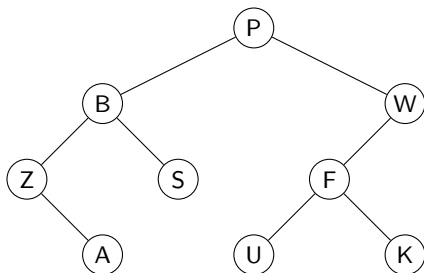
Aufgabe 10



(a) Preorder: P B Z A S W F U K

(b) Inorder: Z A

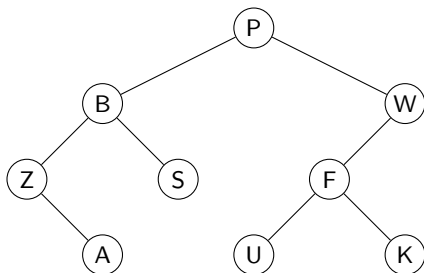
Aufgabe 10



(a) Preorder: P B Z A S W F U K

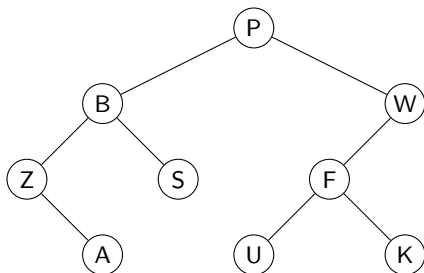
(b) Inorder: Z A B

Aufgabe 10



- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S

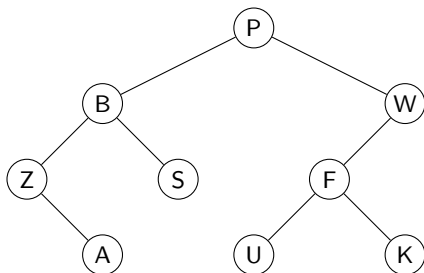
Aufgabe 10



(a) Preorder: P B Z A S W F U K

(b) Inorder: Z A B S P

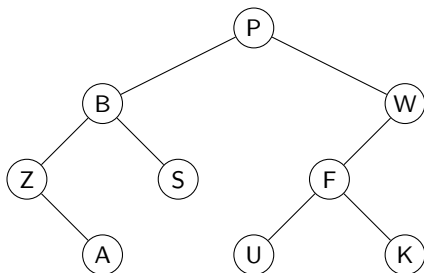
Aufgabe 10



(a) Preorder: P B Z A S W F U K

(b) Inorder: Z A B S P U

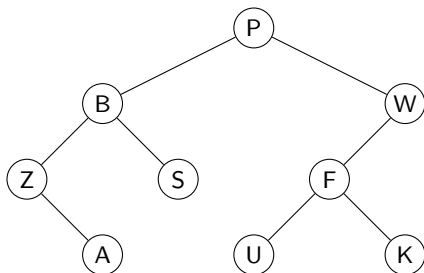
Aufgabe 10



(a) Preorder: P B Z A S W F U K

(b) Inorder: Z A B S P U F

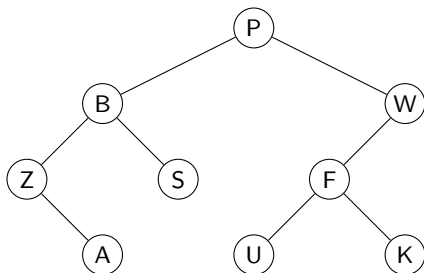
Aufgabe 10



(a) Preorder: P B Z A S W F U K

(b) Inorder: Z A B S P U F K

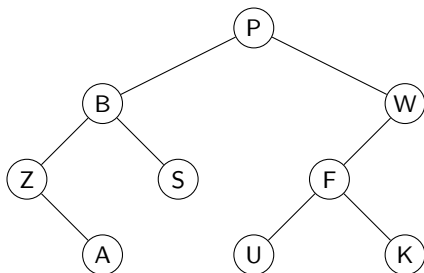
Aufgabe 10



(a) Preorder: P B Z A S W F U K

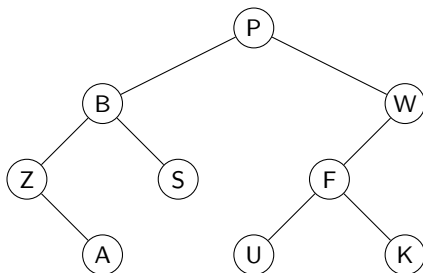
(b) Inorder: Z A B S P U F K W

Aufgabe 10



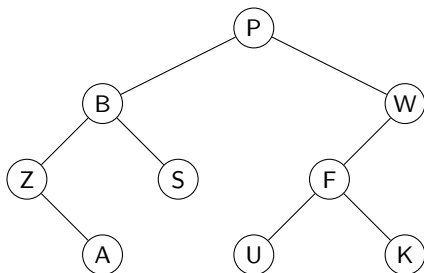
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder:

Aufgabe 10



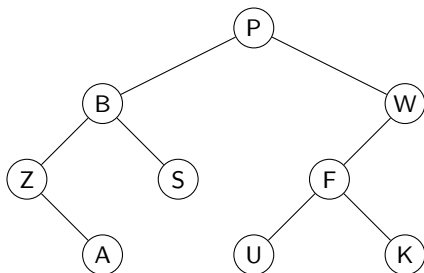
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A

Aufgabe 10



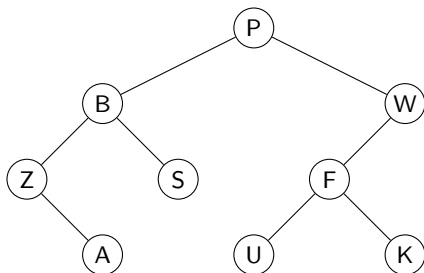
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z

Aufgabe 10



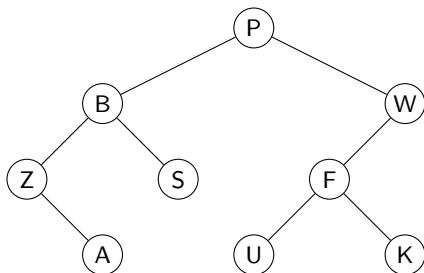
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z S

Aufgabe 10



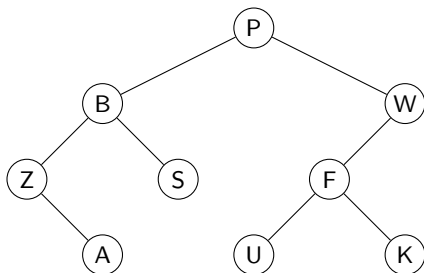
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z S B

Aufgabe 10



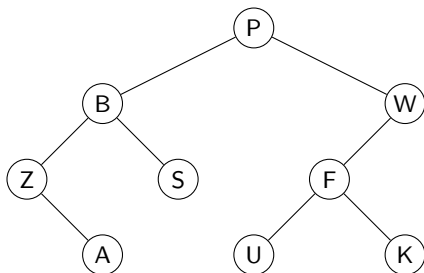
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z S B U

Aufgabe 10



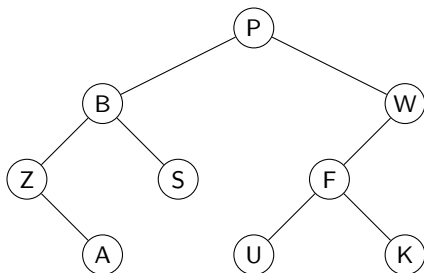
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z S B U K

Aufgabe 10



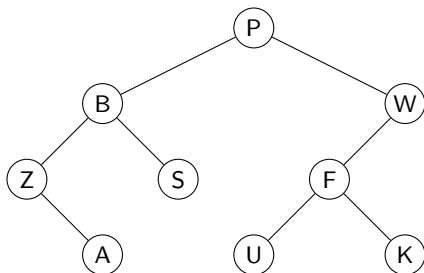
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z S B U K F

Aufgabe 10



- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z S B U K F W

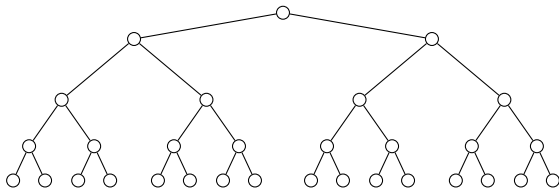
Aufgabe 10



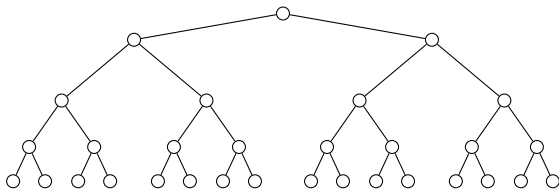
- (a) Preorder: P B Z A S W F U K
- (b) Inorder: Z A B S P U F K W
- (c) Postorder: A Z S B U K F W P

Aufgabe 11

Beschreibe den Typ des Baums möglichst genau mit Fachausdrücken der Informatik.



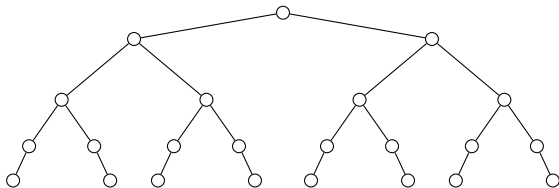
Aufgabe 11



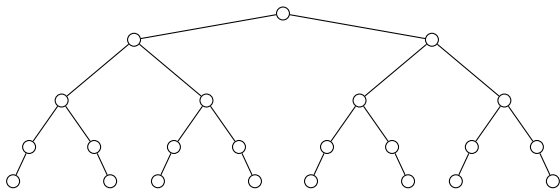
perfekter Binärbaum (binär, weil kein Knoten mehr als zwei Kanten hat)

Aufgabe 12

Beschreibe den Typ des Baums möglichst genau mit Fachausdrücken der Informatik.



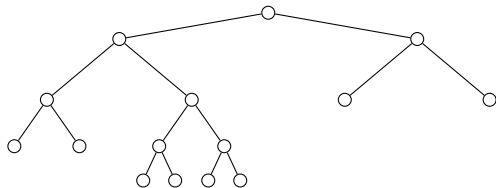
Aufgabe 12



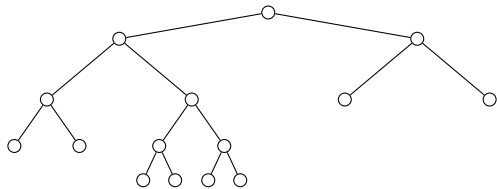
balancierter Binärbaum

Aufgabe 13

Beschreibe den Typ des Baums möglichst genau mit Fachausdrücken der Informatik.



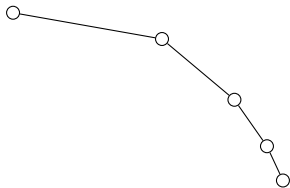
Aufgabe 13



voller Binärbaum

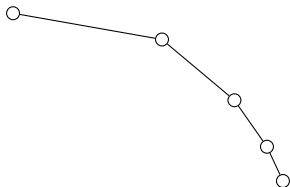
Aufgabe 14

Beschreibe den Typ des Baums möglichst genau mit Fachausdrücken der Informatik.



Aufgabe 14

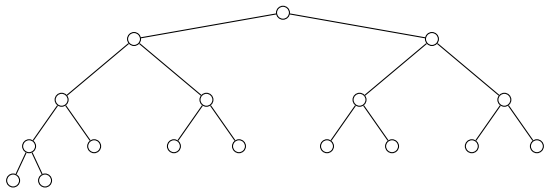
perfekter Binärbaum



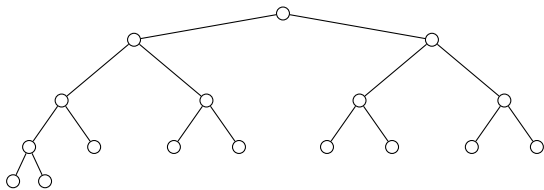
Ein voller Binärbaum, dessen Blätter alle dieselbe Tiefe haben.

Aufgabe 15

Beschreibe den Typ des Baums möglichst genau mit Fachausdrücken der Informatik.



Aufgabe 15



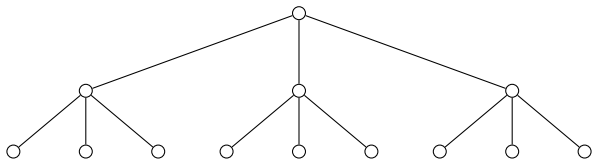
balancierter und voller Binärbaum

Aufgabe 15

Skizziere einen perfekten ternären Baum mit der Höhe 2.

Aufgabe 15

perfekter ternärer Baum der Höhe 2



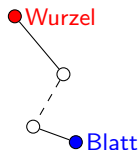
Aufgabe 16

Sind die Aussagen (im Allgemeinen) *wahr* oder *falsch*?

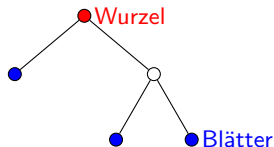
- (a) Ein entarteter Baum hat genau ein Blatt.
- (b) Ein voller Binärbaum hat eine gerade Anzahl Blätter.
- (c) Wenn man in einem Baum eine Kante löscht, erhält man einen Wald.
- (d) Wenn ein Baum die Höhe 3 hat und einer seiner Knoten v die Tiefe 1, dann muss v die Höhe 2 haben.
- (e) Ein perfekter Binärbaum der Höhe 3 hat 14 Kanten.

Aufgabe 16

- (a) Ein entarteter Baum hat genau ein Blatt. *wahr*



- (b) Ein voller Binärbaum hat eine gerade Anzahl Blätter. *falsch*
Gegenbeispiel mit 3 Blättern:



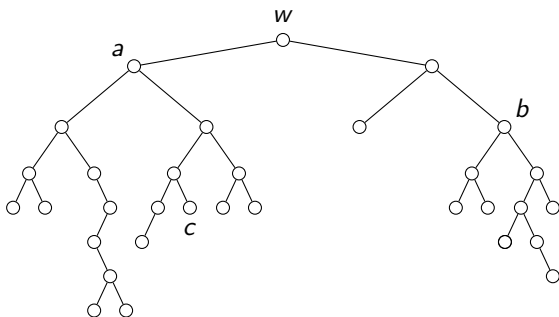
- (c) Wenn man in einem Baum eine Kante löscht, erhält man einen Wald. *wahr*

Beachte, dass ein isolierter Knoten auch ein Baum ist.

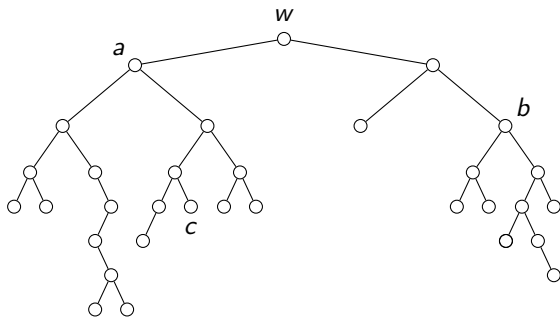
- (d) Wenn ein Baum die Höhe 3 hat und einer seiner Knoten v die Tiefe 1, dann muss v die Höhe 2 haben. *falsch*

Aufgabe 17

Bestimme die Tiefe und die Höhe der angeschriebenen Knoten.



Aufgabe 17



Die *Tiefe eines Knotens* (depth, level) ist die Anzahl der Kanten auf dem Pfad vom Wurzelknoten zum betreffenden Knoten.

Die *Höhe eines Knotens* (height) ist die Anzahl der Kanten entlang des längsten Pfads vom Knoten zu einem Blatt, das im Teilbaum des Knotens liegt.

Die *Höhe eines Baums* (height of a tree) ist die Höhe des Wurzelknotens (Alternativ: Die maximale Tiefe eines Knotens.)

Tiefe(w) = 0, Höhe(w) = 7 = Höhe(*Baum*)

Tiefe(a) = 1, Höhe(a) = 6

Aufgabe 18

Ein Baum wird durch eine mehrfach verschachtelte Liste dargestellt, wobei die erste Komponente den Wert (Key), die zweite Komponente den linken Teilbaum und die dritte Komponente den rechten Teilbaum darstellt. Skizziere mit diesen Angaben folgenden Baum:

$T = [3, [5, [4, [], []], [1, [], []]], [6, [], []]]$

Aufgabe 18

Der Baum kann „algorithmisch“ gezeichnet werden:

Entferne die erste (öffnende) und letzte (schliessend) Klammer

Durchlaufe die Zeichenkette zeichenweise

wenn das Zeichen ein Wert ist:

schreibe ihn auf

wenn das Zeichen eine öffnende Klammer ist:

wenn links noch keine Kante ist:

gehe nach links

sonst:

gehe nach rechts

wenn das Zeichen eine schliessende Klammer ist:

gehe entlang der Kante wieder zurück

wenn es ein anderes Zeichen (Komma, Leerzeichen) ist:

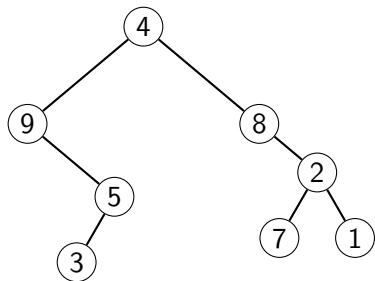
ignoriere es

Zeichne die Kanten ein, die beidseits einen Wert haben

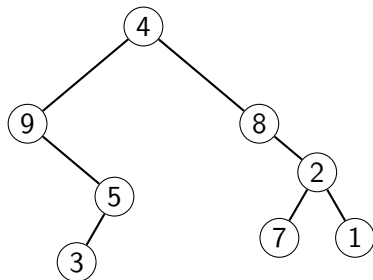
T = [3, [5, [4, [], []], [1, [], []]], [6, [], []]]

Aufgabe 19

Stelle den unten abgebildeten Baum durch eine mehrfach verschachtelte Liste dar. Dabei steht in der ersten Komponente den Wert (Key), in der zweiten Komponente der linken Teilbaum und in der dritten Komponente der rechten Teilbaum.



Aufgabe 19



Die Aufgabe kann ähnlich wie die letzte gelöst werden:

- ▶ Beginne bei der Wurzel.
- ▶ Umfahre den Baum im Gegenuhrzeigersinn.
- ▶ Bei einem Knoten ist der Wert und ein Komma zu notieren.
- ▶ Für jede Bewegung nach unten schreibe [.
- ▶ Fehlt ein linker Knoten, schreibe [] und ein Komma.
- ▶ Fehlt ein rechter Knoten, schreibe [] .
- ▶ Für jede Bewegung nach oben schreibe] und ein Komma.
- ▶ Am Schluss setze [an den Anfang und] ans Ende der

Aufgabe 20

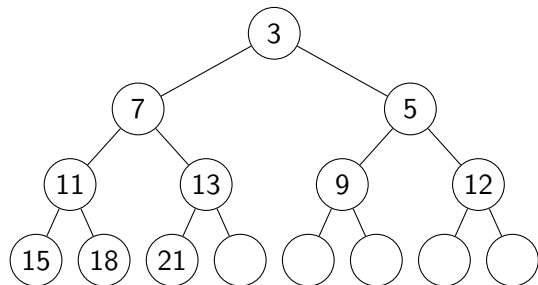
Nenne zwei unterschiedliche Informatik-Anwendungen für Heaps

Aufgabe 20

- ▶ als Vorrangwarteschlange (priority queue)
- ▶ als Datenstruktur für Heapsort

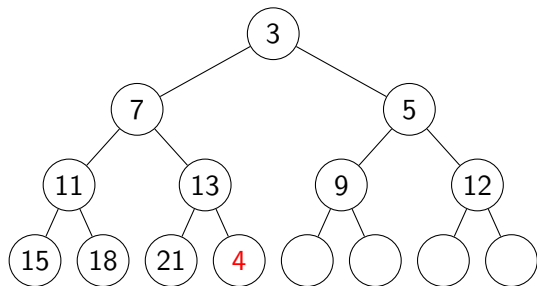
Aufgabe 21

Füge den Knoten mit dem Schlüssel 4 in den Min-Heap ein und zeige schrittweise, wie die Min-Heap-Eigenschaft wieder hergestellt wird. Protokolliere jeden Schritt – oder mindestens die Änderungen – in einem separaten Baum. Ein leeres Baumraster wird an Prüfungen zur Verfügung gestellt.



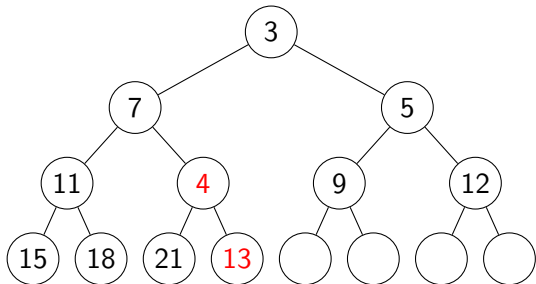
Aufgabe 21

Den neuen Wert 4 am Ende des Heaps einfügen.



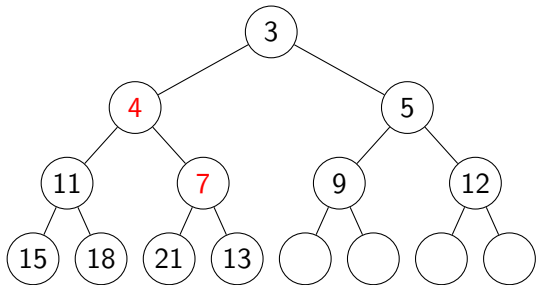
Der Knoten mit dem Wert 4 verletzt die Min-Heap-Eigenschaft.

Eine swim()-Operation mit 4 ausführen.



Der Knoten mit dem Wert 4 verletzt die Min-Heap-Eigenschaft.

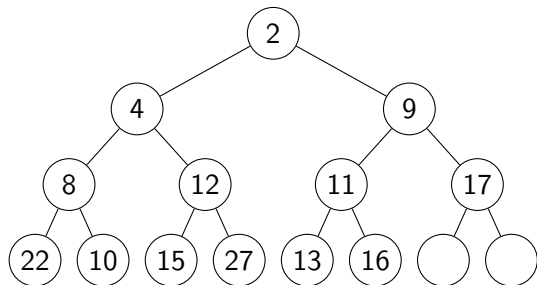
Eine weitere `swim()`-Operation mit 4 ausführen.



Die Min-Heap-Eigenschaft ist wieder hergestellt.

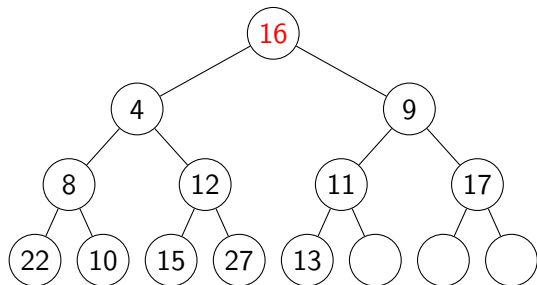
Aufgabe 22

Entferne das kleinste Element aus dem Min-Heap und zeige schrittweise, wie die Heap-Struktur und die Min-Heap-Eigenschaft wieder hergestellt werden. Protokolliere jeden Schritt – oder mindestens die Änderungen – in einem separaten Baum. Ein leeres Baumraster wird an Prüfungen zur Verfügung gestellt.



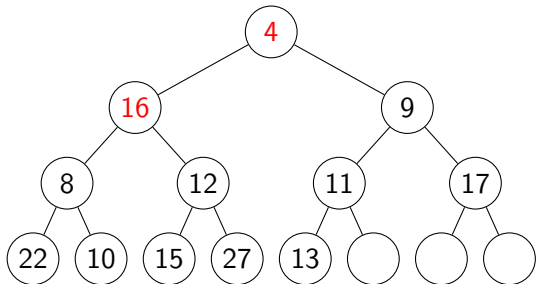
Aufgabe 22

Den letzten Wert (16) in den Wurzelknoten verschieben.



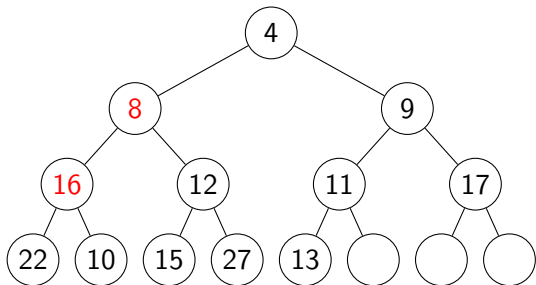
Der Knoten mit dem Wert 16 verletzt die Min-Heap-Eigenschaft.

Mit dem Wert 16 eine sink()-Operation durchführen.



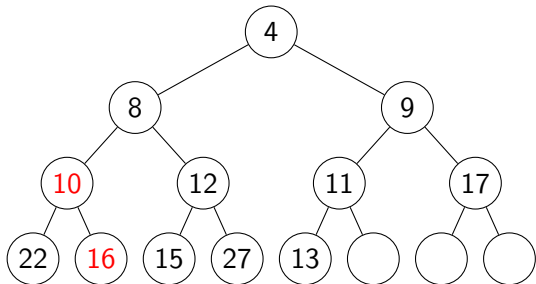
Der Knoten mit dem Wert 16 verletzt die Min-Heap-Eigenschaft.

Mit dem Wert 16 eine weitere `sink()`-Operation durchführen.



Der Knoten mit dem Wert 16 verletzt die Min-Heap-Eigenschaft.

Mit dem Wert 16 eine weitere `sink()`-Operation durchführen.



Die Min-Heap-Eigenschaft ist wieder hergestellt.

Aufgabe 23

Ein Heap besteht aus 68 Knoten.

- (a) Bestimme die Höhe h des Heaps.
- (b) Ein Knoten hat den Listenindex 25. Welchen Index hat sein Elternknoten?
- (c) Ein Knoten hat den Listenindex 34. Welchen Index hat sein rechtes Kind?

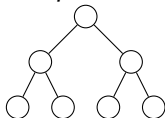
Aufgabe 23

Ein Heap besteht aus 68 Knoten.

- (a) Bestimme die Höhe h des Heaps.

Ein perfekter Baum der Höhe h hat $2^{h+1} - 1$ Knoten.

Beispiel: $h = 2$, $n = 2^3 - 1 = 7$ Knoten



Somit suchen wir $\min \{h \mid 68 \leq 2^{h+1} - 1\} \Rightarrow h = 6$

- (b) Ein Knoten hat den Listenindex 25. Welchen Index hat sein Elternknoten?

$$\lfloor 25/2 \rfloor = 12$$

- (c) Ein Knoten hat den Listenindex 34. Welchen Index hat sein rechtes Kind?

$$2 \cdot 34 + 1 = 69 \text{ [existiert nicht in einem Heap mit 68 Knoten]}$$

Aufgabe 24

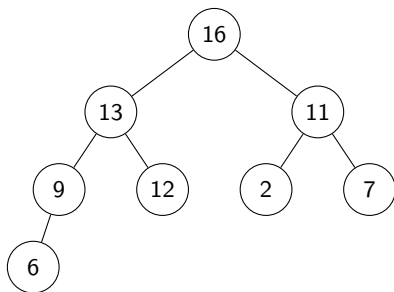
- (a) Stelle den zur folgenden Liste (in Python-Syntax) gehörenden Heap dar.

`H = [None, 16, 13, 11, 9, 12, 2, 7, 6]`

- (b) Handelt es sich beim Heap in (a) um einen Max-Heap? Begründe die Antwort.

Aufgabe 24

(a) $H = [\text{None}, 16, 13, 11, 9, 12, 2, 7, 6]$



(b) Ja, da jeder Elternknoten mindestens so gross ist, wie alle seine direkten Kinder.