

# Prüfungsbesprechung

## Algorithmen & Datenstrukturen 2

Michael Henninger

14. Mai 2023



# Aufgabe 1 - Quiz

- Pro korrekter Antwort: 0.5 Punkte
- Pro falscher Antwort: -0.5 Punkte
- Minimal: 0

## Aufgabe 2 – Binärbaum zeichnen

- 1P: Korrekte Pre-Order Buchstabenreihenfolge
- 1P: Korrekte In-Order Buchstabenreihenfolge

## Aufgabe 3 – Erklären Sie

- 1P pro korrekter Aussage
- - 0.5P für größere Falschaussagen

## Aufgabe 4 – Code Review

- A: Gültiger Bereich im Array bei der Index-Überprüfung
- B: obj und Array-Index an entsprechender Stelle darf beides unabhängig voneinander nicht null sein
- C: Size darf nicht vergrößert werden (Anzahl Elemente bleiben gleich)
- D: Mod-Count darf sich nicht ändern (Struktur ändert sich nicht)

## Aufgabe 5 – AVL Bäume

- a.) und b.): Jeweils 1 Punkt pro korrekter Operation
- c.) 2 Punkte, falls alles korrekt

## Aufgabe 6 – B-Bäume

- a.) und b.): Jeweils 1 Punkt pro korrekter Operation
- c.) 2 Punkte, falls alles korrekt

## Aufgabe 7a – Add Column to Grid

- Für jede der folgenden vergessenen Anpassungen (oder falschen Anpassungen) -0.5 Punkte:
  - 1, 2, 3 am Ende angefügt und innerhalb des Grids richtig verknüpft
  - LastColumn Variable angepasst
  - numColumns Variable angepasst
  - Size angepasst
  - ModCount angepasst



## Aufgabe 7b – toColumn

- A: Funktioniert mit Grid der Länge 1 (1 Punkt)
- B: Funktioniert mit Grid der Länge 2 (2 Punkte)
- C: Funktioniert mit Grid der Länge 12 (2 Punkte)
- Jeweils geprüft:
  - Korrekte Anzahl Knoten in der Rückgabe
  - Korrekte Reihenfolge der Zahlen in der Rückgabe
  - Keine Probleme mit null
  - Alle right-Referenzen der neuen Spalte sind null
- Abzug für (gröbere) Unschönheiten

## Aufgabe 7c – addFirstColumn

- A: Ruft die Methode toColumn genau 1x auf (1 Punkt)
- B: Setzt die Variable firstColumn korrekt (1 Punkt)
- C: Setzt die Variable lastColumn korrekt (1 Punkt)

## Aufgabe 7d – addLastColumn

- A: Ruft toColumn genau 1x auf (0.5 Punkte)
- B: Ändert firstColumn-Variable nicht (0.5 Punkte)
- C: Setzt lastColumn-Variable korrekt (1 Punkt)
- D: Setzt right-Referenzen der zweitletzten Spalte korrekt (2 Punkte)
- E: Komplettes Grid ist korrekt / sonst nichts verändert

## Aufgabe 7e – addColumn

- A: IllegalArgumentException, wenn die Länge der Liste nicht der erwarteten Länge entspricht (0.5 Punkte)
- B: Richtige Hilfsmethode (addFirstColumn oder addAdditionalColumn) aufgerufen (0.5 Punkte)
- C: modCount erhöht (1 Punkt)
- D: size richtig erhöht (nur nicht-null Werte berücksichtigen) (1 Punkt)
- E: Size an passender Stelle erhöht (ohne zusätzlichen Loop) (1 Punkt)
- F: numColumns erhöht (1 Punkt)

# Aufgabe 8a – Update eines Knotenwertes im Binären Suchbaum

- 1 Punkt pro korrekter Zeile

## Aufgabe 8b – validInLeftSubtree

- A: Anker korrekt gesetzt (1 Punkt)
- B: Den «rightmost»-Key für den Vergleich beigezogen (1 Punkt)
- C: Korrekter vergleich (< und nicht «zu viel / anders (unnötiges)» geprüft) (1 Punkt)

## Aufgabe 8c – validInRightSubtree

- A: Anker korrekt gesetzt (1 Punkt)
- B: Den «leftmost»-Key für den Vergleich beigezogen (1 Punkt)
- C: Korrekter vergleich (> und nicht «zu viel / anders (unnötiges)» geprüft) (1 Punkt)

## Aufgabe 8d – validInParents

- A: Gibt true zurück, wenn Parent null ist (1 Punkt)
- B: Prüft, ob `update > als parent.key`, wenn `original > parent.key` (1 Punkt)
- C: Prüft, ob `update < parent.key`, wenn `original < parent.key` (1 Punkt)
- D: Rekursion von `validateInParents` (1 Punkt)
- E: Funktioniert einwandfrei (1 Punkt)



# Aufgabe 8e – Vorzeitiger Abbruch

- Ist möglich. 1 Punkt für korrekte Begründung