

Aufgabe 1 (Alida Fettel 5)

Max Kahl
Marvin Kint
Konstantin Neunthaler

— Binär Suchbäume

(c)

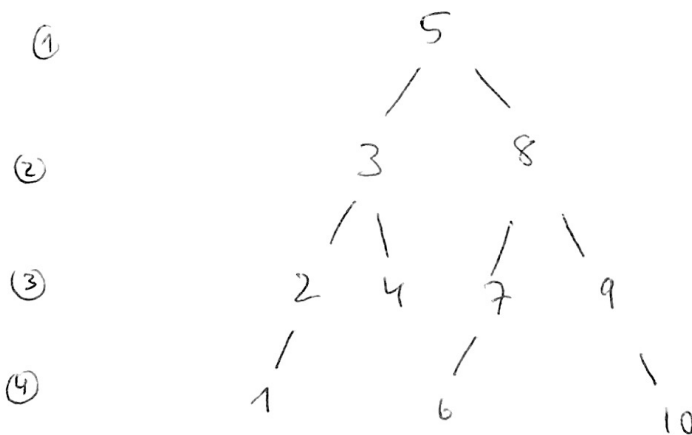
Um eine möglichst geringe Tiefe des Suchbaums zu erreichen, wäre es optimal immer das mittlere Element einer (Unk-)Liste einzufügen. Nach Einfügen des mittleren Elements teilt man die übrigen Daten wiederum in linke und rechte Unterliste und fügt das mittlere Element diese Listen als node-left bzw. node-right ein.

Bsp: key-Datensatz: key-list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

einfügen: $\underbrace{5}_{(1)} \rightarrow \underbrace{3, 8}_{(2)} \rightarrow \underbrace{2, 4, 7, 9}_{(3)} \rightarrow \underbrace{1, 6, 10}_{(4)}$

depth
des nodes

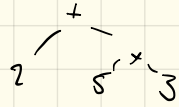
Resultat: (1)



(d) _____

②

b) • $2 + 5 \times 3 \approx 253 \times + \approx$



• $249 \approx (3 + (4 - 7) * 8) - (1 - 6) \approx 24 \times 347 - 844 \times 16 - -$

\approx



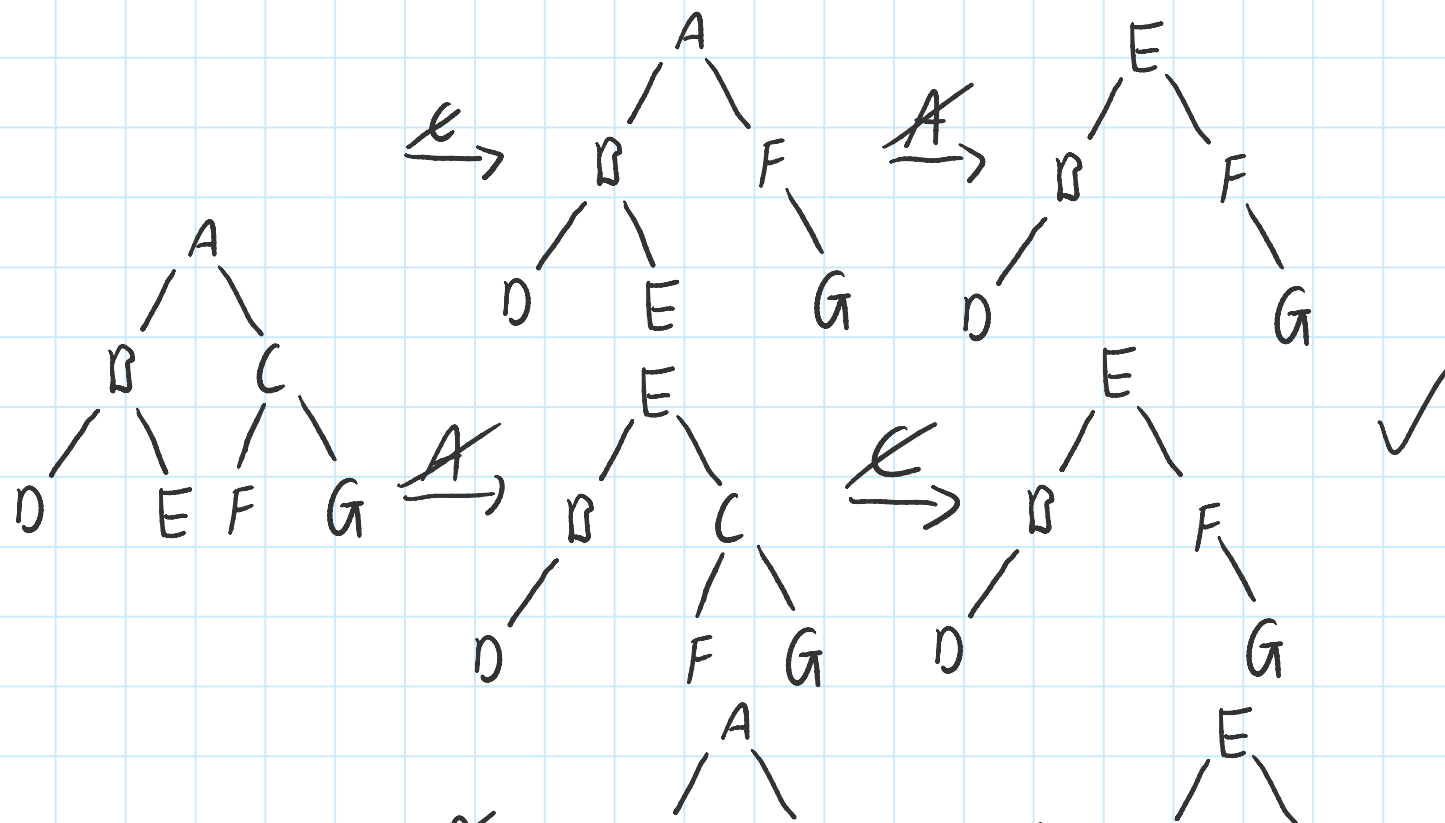
Blatt 5 Nummer 1d) (nur digitale Abgabe)

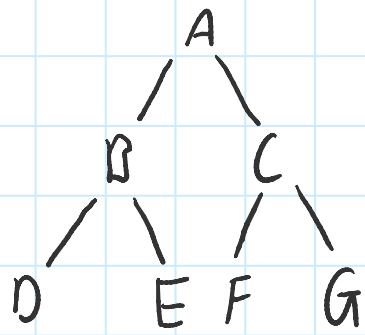
Dienstag, 28. Mai 2019 15:51

Fall 1: Nur Löschung nodes ohne Kinder \Rightarrow voneinander unabh. \Rightarrow kommutativ

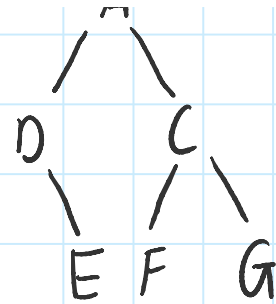
Fall 2: Löschung Kind und Elter \Rightarrow von drei bleibt nur einer übrig \Rightarrow kommutativ

Fall 3: Löschung mehrerer Eltern (geeignet für niedrigste Ordnung, höhere ändert nichts)

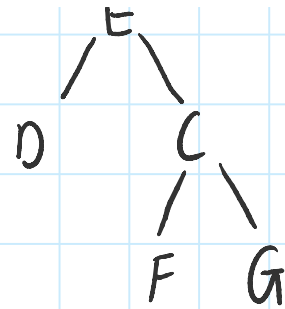




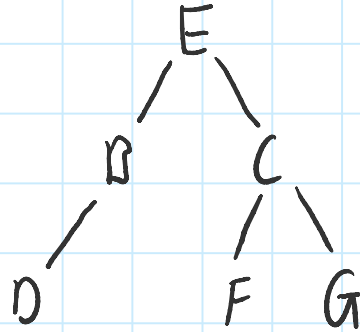
~~B~~ →



~~A~~ →



~~A~~ →



~~B~~ →

