

Übungsblatt 8

Truong, Debel

Aufgabe 3

a. Red-Black-Tree

| Name | Runtime |
|---------------|-----------------------|
| Search | $\mathcal{O}(\log n)$ |
| Failed Search | $\mathcal{O}(n)$ |
| Insert | $\mathcal{O}(\log n)$ |
| Delete | $\mathcal{O}(\log n)$ |

b. verketteter Liste. Allerdings werden die Schlüssel innerhalb jeder Verkettung in sortierter Reihenfolge gehalten.

| Name | Runtime |
|---------------|--|
| Search | $\mathcal{O}(n)$ |
| Failed Search | $\mathcal{O}(\frac{n}{m} + 1)$ (m: hashtable size) |
| Insert | $\mathcal{O}(n)$ |
| Delete | $\mathcal{O}(n)$ |

Aufgabe 4

a. Hashfunktion:

$$h : I \rightarrow \{0, \dots, m-1\}$$

$$h(k) = k \bmod m$$

Surjektivität: Für alle $k > m$ existiert $h(k) \in \{0, \dots, m-1\}$, da $h(k) = k \bmod m$

b. b : Binary Search Tree with n keys

Alle Schlüssel s in b gilt: $s \in \{1, 2, \dots, 99\}$

Hashfunktion:

$$f : I \rightarrow \{1, \dots, n\}$$

$$f(s) = (n \bmod 99) + 1$$

z.B:

Minimum beispiel: $f(99) = (99 \bmod 99) + 1 = 1$, also key ist 1 und value ist 99

$f(1) = (1 \bmod 99) + 1 = 2$, also key ist 2 und value ist 1

Maximum beispiel $f(98) = (98 \bmod 99) + 1 = 99$, also key ist 99 und value ist 98