

Aufgabe: Aufwandsabschätzung

Warum ignorieren wir bei der Aufwandsabschätzung konstante Faktoren, die Problemgröße (= Anzahl der Elemente) aber nicht?

- A: Um Unabhängigkeit von Hardware, Compiler, etc. zu erreichen, aber trotzdem aussagekräftige Ergebnisse zu erhalten.
- B: Weil das eine sehr einfache Faustregel ist, die man sich leicht merken kann.
- C: Weil es nur darauf ankommt, zwecks Vergleichbarkeit stets die gleichen Kriterien anzuwenden. Welche Kriterien das sind, ist eigentlich egal.
- D: Um es nicht zu leicht zu machen. Wir brauchen ja gute Prüfungsfragen.

Aufgabe: Hash-Funktion

Warum soll `hashCode` für unterschiedliche Objekte möglichst unterschiedliche Ergebnisse liefern?

- A: Damit nicht zu viel passiert, wenn man `hashCode` fälschlicherweise für Objektvergleiche einsetzt.
- B: Damit alle Listen in der Hashtabelle etwa gleich kurz bleiben.
- C: Damit die Hash-Werte einfacher zu berechnen sind.
- D: Um Größenvergleiche zwischen Hash-Werten zu ermöglichen.

Aufgabe: Hash-Funktion

Die binäre Suche ist sehr effizient. Warum verwenden wir dennoch häufig Datenstrukturen wie Bäume und Hashtabellen?

- A: Weil die Suche in Bäumen und Hashtabellen noch viel effizienter ist.
- B: Weil nicht alle Elemente vergleichbar sind oder einen Hash-Wert haben.
- C: Weil das Einfügen in ein sortiertes Array sehr aufwendig ist.
- D: Weil wir mehrere unterschiedliche Datenstrukturen kennenlernen wollen.