

Prof. Dr. Hans-Peter Kriegel  
Thomas Bernecker, Tobias Emrich

Übungen zur Vorlesung  
**Effiziente Algorithmen**

**Aufgabe 6.1: Hashfunktionen (3+3 Punkte)**

Gegeben seien  $2^8$  Buckets und 4-stellige Zahlen als Schlüssel. Es sollen die Schlüssel 2040 bis 2050 einsortiert werden.

- Implementieren Sie eine Hashfunktion nach der Middle-Square Methode in Java. Welche Werte liefert diese Hashfunktion für die gegebenen Schlüssel?
- Implementieren Sie eine Hashfunktion nach der Divisionsmethode in Java. Verwenden Sie für die Division eine möglichst gut geeignete Primzahl (unabhängig von den gegebenen Schlüsseln!). Begründen Sie Ihre Wahl! Welche Werte liefert diese Hashfunktion für die gegebenen Schlüssel?

**Aufgabe 6.2: Offenes Hashing (2+2 Punkte)**

Gegeben sei die Hashfunktion  $h(k) = k \text{ MOD } 7$  mit der folgenden Hashtabelle:

0	1	2	3	4	5	6

sowie die folgenden numerischen und nicht-numerischen Schlüssel:

a) 1, 7, 13, 15, 14, 5

b) a, k, v, d, y

Diese Schlüssel sollen nun mittels offener Divisionsmethode in die Hashtabelle einsortiert werden. Geben Sie jeweils die gefüllten Hashtabellen an!

**Aufgabe 6.3: Hashing mit geschlossener Kollisionsstrategie (3+1 Punkte)**

Es sei die Hashfunktion gegeben, die für einen Schlüssel die zugehörige Adresse berechnet. Zur Kollisionsbehandlung verwende man die geschlossene Kollisionsstrategie: direkte Verkettung mit Verschmelzen.

- Tragen Sie die Schlüsselfolge 5, 24, 9, 46, 1, 35, 31, 10, 53 mit in eine Tabelle der Größe 11 ein! Zeichnen Sie die Hashtable dabei nach der 3., 5., 6. und letzten Einfügung!
- Vergleichen Sie die beiden geschlossenen Kollisionstrategien direkte Verkettung mit Verschmelzen und lineares Sondieren ('linear probing') miteinander! Welche Vorteile hat jede gegenüber der jeweils anderen Strategie?