

## Dokumentation – Hashtable

### Aufwandsvergleich:

	Array	Linked List	Hashtable
Einfügen	$O(1)$	$O(1)$	$O(1)$
Löschen	$O(1)$	$O(1)$	$O(1)$
Suchen	$O(n)$	$O(n)$	$O(n)$

Alle Angaben beziehen sich auf den Worstcase. Bei der Linked List wurde bezüglich des Löschens und Einfügens immer das Erste Element genommen. Wenn ein speziellen Element gelöscht oder sortiert eingefügt werden soll, dann ist beides  $O(n)$  im Worstcase.

Bei der Hashtable hängt der Aufwand zum Suchen eines Elements stark von der gewählten Hashfunktion, sowie dem Füllgrad ab. Der Aufwand steigt indirekt proportional mit dem Füllgrad, da die Wahrscheinlichkeit für Kollisionen mit erhöhtem Füllgrad wächst. Im Idealfall, also wenn keine Kollisionen passieren, ist der Aufwand beim Suchen in einer Hashtabelle  $O(1)$ . Im Worstcase  $O(n)$ .

### Hashfunktion:

Für die Hashfunktion wurde die hash-Code Methode aus Java herangezogen.

$$h(s) = s[0] * 31^{(n-1)} + s[1] * 31^{(n-2)} \dots + s[n-1]$$

$S[x]$ ... Wert des Zeichens laut Ascii-Tabelle an der Stelle  $x$  im String  $s$

$n$ .. Länge des Strings  $s$

$h(s)$ ... Hashwert

### Kollisionserkennung:

Im Falle einer Kollision wird die quadratische Sondierung angewendet. Es wird also ausgehend vom berechneten Hashindex, der bereits belegt ist, im Quadrat der natürlichen Zahlen (1,4,9,...) weitergezählt bis ein freier Platz gefunden ist. Dabei wird der neu berechnete Index modulo Array-Größe gerechnet, sodass falls er außerhalb des Arrays kommen sollte, von vorne weiter macht. Elemente mit dem key „Sentinel“ werden dabei ebenso behandelt wie Elemente mit „None“, da es sich bei den Sentinel-Elementen um gelöschte Elemente handelt und auf Grund der quadratischen Sondierung nicht auf Empty gesetzt werden können.

### Verwaltung der Kursdaten:

Beim Import von Kursdaten werden die als verschachtelte Liste gespeichert. Jedes Element der äußeren Liste enthält genau 1 Datum mit den entsprechenden Werten (siehe Angabe).

### Löschalgorithmus:

Beim Löschen wird das zu löschende Element gesucht. Ausgehend vom berechneten Hashwert wird in quadratischen Schritten (1,4,9,...) weitergesucht solange der Key nicht übereinstimmt. Sobald man auf ein leeres Element stößt, wird der Vorgang abgebrochen, da das gesuchte Element nicht in der Liste ist. Beim eigentlichen Löschen wird der Key auf „Sentinel“ gesetzt, da auf Grund der quadratischen Sondierung Elemente für Operationen ausgehend von dem selben Hash bearbeitet werden müssen.

### Programmaufruf:

Zum Aufrufen in den Projektordner wechseln. „Python main.py“ in der Kommandozeile eingeben. Folgende Module müssen am PC installiert sein:

```
import csv
from tabulate import tabulate
import json
from matplotlib import pyplot as plt
from datetime import datetime as dt
```