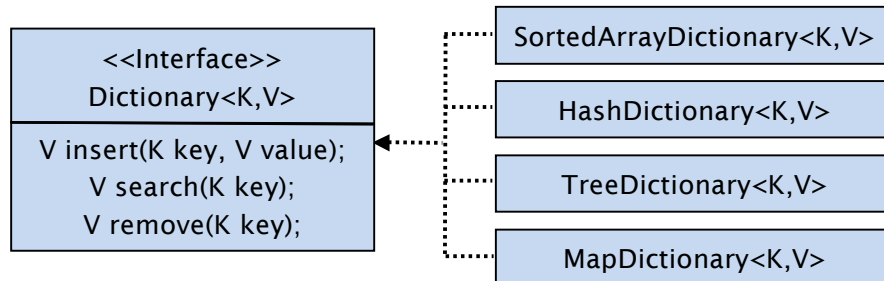


## Aufgabenblatt 1



- Das Interface *Dictionary* (siehe Web-Seite) definiert Methoden zur Verwaltung von Datensätzen, die aus einem Schlüssel vom Typ K und Nutzdaten vom Typ V bestehen.
- `SortedArrayDictionary` implementiert ein Dictionary mit einem Feld, in dem die Datensätze lückenlos und sortiert gespeichert werden. Für die Suche wird binäre Suche eingesetzt.
- `HashDictionary` enthält als Implementierung eine wie in der Vorlesung besprochene Hashtabelle mit linear verketteten Listen.
- `TreeDictionary` setzt die in der Vorlesung besprochenen AVL-Bäume ein.
- `MapDictionary` benutzt ein Map-Objekt aus den Java-Collections. Das Map-Objekt wird dem Konstruktor als Parameter übergeben. In der Anwendung ist sowohl ein `HashMap`- als auch ein `TreeMap`-Objekt zu übergeben.

Es sind folgende Aufgabenteile zu lösen:

1. Implementieren Sie alle Klassen in Java und testen Sie die Klassen ausgiebig. Alle Klassen sollen die `toString`-Methode in geeigneter Weise überschreiben.
2. Schreiben Sie eine GUI für eine Wörterbuch-Anwendung Deutsch-Englisch mit folgenden Funktionalitäten (siehe dazu letzte Aufgabe aus Fortgeschrittene Programmieretechnik):
  - a. Auswahl der Implementierung (`ArrayDictionary`, `HashDictionary`, etc.) über einen Menüeintrag.
  - b. Auswahl einer Datei mit Deutsch-Englisch-Wörtern und Einlesen der Datei über einen Menüeintrag. Ein Wörterbuch-Datei mit ca. 16000 Einträgen finden Sie auf der Web-Seite.
  - c. Nachschlagen eines deutschen Wortes.
  - d. Löschen eines Wortpaares.
  - e. Einfügen eines Wortpaares.
  - f. Ausgeben des Wörterbuchs in einem Ausgabefeld.
3. Die verschiedenen Implementierungen sollen einer Performance-Untersuchung unterzogen werden. Messen Sie dazu die CPU-Zeiten für verschiedene Anwendungsfälle für die gegebene Wörterbuchdatei mit ungefähr  $n = 16000$  Einträgen und tragen Sie die Zeiten in folgende

Tabelle ein. Die Performance-Untersuchung soll ebenfalls über die GUI gestartet werden können.

	SortedArray -Dictionary	Hash- Dictionary	Tree- Dictionary	HashMap- Dictionary	TreeMap- Dictionary
Aufbau eines Wörterbuchs mit <b>n = 8000</b> Einträgen					
Aufbau eines Wörterbuchs mit <b>n = 16000</b> Einträgen					
Erfolgreiche Suche <sup>1)</sup> für <b>n = 8000</b>					
Erfolgreiche Suche <sup>1)</sup> für <b>n = 16000</b>					
Nicht erfolgreiche Suche <sup>2)</sup> für <b>n = 8000</b>					
Nicht erfolgreiche Suche <sup>2)</sup> für <b>n = 16000</b>					

<sup>1)</sup> Bei der erfolgreichen Suche wird jedes deutsche Wort des Wörterbuchs genau einmal gesucht und die Gesamtzeit gemessen. Testen Sie mit einem Wörterbuch mit  $n = 8000$  und dann mit  $n = 16000$  Einträgen.

<sup>2)</sup> Bei der nicht erfolgreichen Suche kann einfach nach jedem englischen Wort gesucht werden, das ja als deutsches Wort so gut wie nicht vorkommen kann. Auch hier ist die Gesamtzeit für  $n = 8000$  bzw.  $n = 16000$  einzutragen.