

1 Binärsuche

Erstellen Sie eine Klasse `de.hska.iwi.ads.solution.search.BinarySearch`, die das Interface `de.hska.iwi.ads.search.Search` implementiert.

Es soll eine Variante der Binärsuche implementiert werden, die wie folgt einen Index i für die Suche nach einem Schlüssel key in einem Feld $a[left..right]$ zurück gibt:

- Falls key in $a[left..right]$ enthalten ist, dann soll i die Position von key in a sein: $key = a[i]$. Bei doppelten Vorkommen von Werten soll der niederwertigste Index (am weitesten links stehende) zurückgegeben werden. Es darf nicht sequentiell in diesem Fall gesucht werden.
- Falls key in $a[left..right]$ nicht enthalten ist, dann gibt es die folgenden drei Fälle:
 1. $key < a[left]$: Dann soll $i = left - 1$ sein.
 2. $key > a[right]$: Dann soll $i = right + 1$ sein.
 3. $a[left] \leq key \leq a[right]$: Dann soll i der kleinste Index sein, so dass $a[i - 1] < key < a[i]$ gilt.

Der Zeitaufwand soll im schlimmsten Fall $\Theta(\log_2 n)$ sein. Sie dürfen also in keinem Fall eine sequentielle Suche durchführen. Vor allem nicht bei einem Feld mit gleichen Werten.

Hinweise:

- Die beiden Sonderfälle 1 und 2 können Sie am besten vor einer herkömmlichen Binärsuche mit Größenvergleiche entdecken und behandeln.
- Falls $key \notin a$ gilt, bricht eine herkömmliche Implementierung der Binärsuche üblicherweise ab, wenn $right < left$ gilt. Bei fast allen Implementierungen gilt dann $left = right + 1$. Der letzte Fall 3 kann in einer Nachverarbeitung einer herkömmlichen Binärsuche entdeckt und behandelt werden.

Es ist deswegen sinnvoll, wenn Sie zuerst eine „normale“ Binärsuche, wie in der Vorlesung behandelt, implementieren und testen (iterativ oder rekursiv). Diese Implementierung kann durch Methodenaufruf wiederverwendet werden. Diese Methode sollte in der endgültigen Version `private` deklariert sein.

Benutzen Sie die abstrakte Testklasse `de.hska.iwi.ads.search.SearchTest` als Basisklasse für Ihre JUnit-Testklasse. Fügen Sie eigene Test zu Ihrer Testklasse hinzu, um Fehler in Ihrer Implementierung zu finden.