

Aufgabenblatt 7

Ein bitonisches Array ist eine Folge von Zahlen, die zunächst streng monoton steigt und dann streng monoton fällt. In dieser Aufgabe sollen Sie sowohl eine iterative als auch eine Teile-und-herrsche-Lösung zum Finden des größten Wertes - des "Gipfelwertes" - eines bitonischen Arrays beliebiger Länge implementieren, und die Laufzeiten der beiden Lösungen messen und analysieren. Gehe Sie dazu wie folgt vor:

- Implementieren Sie eine iterative Lösung des Problem. Ihre Lösung sollte die folgende Signatur haben:

```
public static int findPeakIterative(int[] a)
```

Der Rückgabewert soll der Index des Gipfelwertes sein.

- Implementieren Sie eine Teile-und-Herrsche-Lösung des Problem. Diese Methode sollte die folgende Signatur haben:

```
public static int findPeakDivideAndConquer(int[] a)
```

Der Rückgabewert soll der Index des Gipfelwertes sein.

Hinweis: Sie benötigen eine private rekursive Methode, die neben dem Parameter `a` auch die linke und die rechte Grenze des zu betrachtenden Arrays übergeben bekommt.

- Testen Sie ihre Methoden und messen sie deren Laufzeiten mit Hilfe der bereits vorhandenen Klasse `BitonicArrayTest`.
- Analysieren Sie die Laufzeitkomplexität der iterativen und der Teile-und-Herrsche-Lösung in der O-Notation.
- Zeigen Sie, ob bzw. dass die gemessenen Laufzeiten Ihre Analyse bestätigen.
- Welche Laufzeiten würden Sie, auf Basis Ihrer Komplexitätsanalyse und der gemessenen Zeiten, für Arrays der Länge 1.000.000.000 erwarten?