

Übungsblatt 8 zur Vorlesung Programmieren

Ausgabe: Fr 29.12.2023 – Abgabe und Abnahme bis Do 18.01.2024

Aufgabe 1 (Rekursive Berechnung der Fibonacci-Folge, Caching, Laufzeitmessung):

Die Fibonacci-Folge (F_i) haben wir in der Vorlesung für $i \geq 0$ definiert durch

$$F_0 = 0, \quad F_1 = 1, \quad F_i = F_{i-1} + F_{i-2}.$$

Außerdem haben wir zwei Berechnungsverfahren betrachtet, ein iteratives und ein rekursives.

- a) Implementieren Sie die beiden Berechnungsverfahren in einer Klasse `Fibonacci` in den Methoden

```
public static int fibonacciIter(int n) und  
public static int fibonacciRec(int n),
```

die jeweils das n -te Element der Fibonacci-Folge berechnen sollen.

- b) Fügen Sie der `Fibonacci`-Klasse eine `main`-Methode hinzu, welche eine Zahl n von der Kommandozeile einliest und dann die n -te Fibonacci-Zahl mit beiden Verfahren berechnet. Bis zu welcher Zahl n erhalten Sie korrekte Ergebnisse?

- c) Legen Sie nun eine zweite Klasse `FibonacciBigInteger` an. Implementieren Sie darin die beiden Methoden

```
public static BigInteger fibonacciIter(int n) und  
public static BigInteger fibonacciRec(int n),
```

die zur Berechnung von Fibonacci-Zahlen den Datentyp `BigInteger` anstelle von `int` verwenden sollen. Fügen Sie analog zu Aufgabenteil b) eine `main`-Methode hinzu.

- d) Fügen Sie nun Ihrer rekursiven Implementierung einen Cache für bereits berechnete Folgenglieder hinzu (analog dem Cache für Binomialkoeffizienten aus der Vorlesung). Erstellen Sie dazu eine weitere Methode `public static BigInteger fibonacciRecCache(int n)`.

- e) Gibt es eine Grenze für n , oberhalb derer die rekursive Version mit Cache nicht mehr richtig arbeitet?

- f) Führen Sie Zeitmessungen durch, um die iterative Version und die rekursive mit Cache hinsichtlich Ihrer Laufzeit zu vergleichen. Verwenden Sie die `System.nanoTime()`-Methode zur Zeitmessung, die einen Zeitstempel in Nanosekunden-Genauigkeit als `long`-Wert zurückgibt. Geben Sie die Laufzeiten der beiden Berechnungsverfahren in Millisekunden mit drei Nachkommastellen aus.

- g) Finden Sie heraus, wie Sie den Stack-Speicher in der Java Virtual Machine vergrößern können.