

Aufgabe 8.1

Im Material zu diesem Übungsblatt ist die Klasse `MergeSort.java` enthalten. Diese ist eine Erweiterung der Klasse gleichen Namens, die wir in der Vorlesung besprochen haben. In dieser Aufgabe soll der in der Vorlesung angesprochene verschwenderische Umgang mit dem Arbeitsspeicher genauer untersucht werden. Dazu beschränken wir den zur Ausführung des Programms zur Verfügung stehenden Arbeitsspeicher auf 10 Megabyte:

```
java -Xmx10M MergeSort 10000
```

Dieser Aufruf des Programms erzeugt eine zufällige Eingabefolge mit 10000 Elementen und sollte ohne Laufzeitfehler ausgeführt werden können.

- (a) Was ist die maximale Länge der Eingabefolge, für die die zur Verfügung gestellten 10 Megabyte gerade noch ausreichen, um die Eingabefolge zu sortieren? (1 Punkt)
- (b) Optimieren Sie die Implementierung des MergeSort-Algorithmus aus der Vorlesung so, dass auch noch längere Eingabefolgen verarbeitet werden können. Was ist die maximale Länge der Eingabefolge, die Ihre optimierte Version des Programms sortieren kann, wenn man den verfügbaren Arbeitsspeicher auf 10 Megabyte beschränkt? (5 Punkte)

Aufgabe 8.2

Im Material zur Übung ist zudem die Klasse `ReCurseMe.java` enthalten. Diese enthält die rekursive Methode `g`, die eine Funktion $g : \mathbb{N} \rightarrow \mathbb{Z}$ berechnet.

- (a) Entnehmen Sie dem Programmcode, wie die Funktion g genau definiert ist, und schreiben Sie diese als Definition mit Fallunterscheidung auf. (1 Punkt)
- (b) Wenn man die Methode `g` ausführt, stellt man fest, dass die benötigte Rechenzeit sehr schnell mit dem übergebenen Parameter n wächst. Woran liegt das? Wie würden Sie die Berechnung der Funktion g implementieren, sodass sich die Rechenzeit wesentlich verringert? (4 Punkte)

Aufgabe 8.3

Für jede positive gerade natürliche Zahl n betrachten wir die Folge

$$F_n = f_0, f_1, f_2, \dots, f_{n-1},$$

die definiert ist durch

$$f_i = \begin{cases} \frac{n}{2} + \frac{i+2}{2} & \text{falls } i \text{ gerade,} \\ \frac{n}{2} - \frac{i-1}{2} & \text{falls } i \text{ ungerade.} \end{cases}$$

Was ist die Anzahl der Durchläufe der äußeren `while`-Schleife bei BubbleSort für die Eingabefolge F_n ? (2 Punkte)
