

Name: _____

Matrikelnummer: _____

Aufgabe 1

Mehrere Aufträge sollen so auf zwei gleichartige Maschinen verteilt werden, dass die Aufträge möglichst schnell fertig gestellt werden. Dabei kann ein einzelner Auftrag nicht geteilt (also auf beiden Maschinen) bearbeitet werden. Schreiben Sie eine Klasse *Verteilung* mit der Methode

double optimaleBearbeitungszeit(double[] bearbeitungszeiten),
die die optimale Gesamtbearbeitungszeit der Aufträge zurückliefert. Die Bearbeitungszeiten der einzelnen Aufträge werden der Methode in einem Array übergeben. Die Bearbeitungszeit auf einer Maschine ist die Summe der Bearbeitungszeiten der Aufträge, die dieser Maschine zugeteilt werden. Die Gesamtbearbeitungszeit der beiden Maschinen ist die größere Bearbeitungszeit der beiden Maschinen.

Hinweis: Die Aufgabe kann mittels Exhaustive Search durch eine rekursive Methode gelöst werden.

Aufgabe 2

Schreiben Sie eine Klasse *Berechnung* mit der Methode

int f(int k),

die für $k \geq 1$ folgende Funktion f berechnet:

- $f(1) = 0$.
- Wenn k gerade, dann $f(k) = f(k/2) + 1$.
- Wenn k ungerade, dann $f(k) = f(3k-1) + 1$.

Die Klasse *Berechnung* soll weiters eine Methode

boolean checkF(k,n)

enthalten, die für $k \geq 1$ *true* genau dann zurückliefert, wenn $f(k) \leq n$. Die Methode *checkF(k,n)* soll auch dann *false* zurückliefern, wenn $f(k) = +\infty$, d.h. wenn die Rekursion für f nicht terminiert. Das ist möglich, da die Rekursion für f evt. keinen Fortschritt in Richtung Basisfall macht.

Hinweis: Die Methode *checkF()* kann, analog zur Methode f , recht einfach rekursiv implementiert werden.