

Name: \_\_\_\_\_

Matrikelnummer: \_\_\_\_\_

## Aufgabe 1

Eine Anzahl von Aufträgen soll möglichst gleichmäßig auf Maschinen verteilt werden. Jeder Auftrag kann auf einer beliebigen der Maschinen gefertigt werden, allerdings muss ein Auftrag immer zur Gänze auf der gewählten Maschine gefertigt werden. Die Maschinen arbeiten mit unterschiedlicher Geschwindigkeit und die Aufträge sind von unterschiedlicher Größe. Für eine gleichmäßige Aufteilung der Aufträge sollen die Gesamtarbeitszeiten der Maschinen möglichst gleich sein. Die Arbeitszeit für einen Auftrag auf einer bestimmten Maschine ergibt sich als *groesse/geschwindigkeit*.

Eine optimale Lösung ist nur sehr aufwendig zu berechnen, daher soll der folgende vereinfachte Algorithmus implementiert werden: Nachdem die Aufträge absteigend nach ihrer Größe sortiert wurden, werden die Aufträge in dieser Reihenfolge jeweils der – inklusive das aktuellen Auftrags – am wenigsten belasteten Maschine zugewiesen.

*Beispiel:* Für die Auftragsgrößen 10,6,4,3 und den Maschinengeschwindigkeiten 1,1,2 ergeben sich die Zuweisungen

M1: 6; M2: 4; M3: 10,3.

(Vor der Zuweisung von Auftrag 3 sind die Gesamtarbeitszeiten der Maschinen 6, 4, und 5. Würde Auftrag 3 M2 zugewiesen, wäre deren Gesamtarbeitszeit 7, was größer als die Gesamtarbeitszeit 6.5 für M3 ist).

Gegeben ist die Klasse *Maschine* mit der Methode

```
public double getGeschwindigkeit(),
```

und die Klasse *Auftrag* mit der Methode

```
public double getGroesse().
```

Schreiben Sie die Klasse *Scheduler* mit der Methode

```
Map<Auftrag,Maschine> weiseZu(Auftrag[] auftraege, Maschine[] maschinen),
```

die die Zuweisung der Aufträge zu Maschinen zurückliefert. Sie können annehmen, dass die Aufträge bereits absteigend nach ihrer Größe sortiert sind.

## Aufgabe 2

Für  $n$  Studierende stehen genau  $n$  Studienplätze an verschiedenen Standorten zur Verfügung. Die Studienplätze werden von den Studierenden jeweils unterschiedlich bewertet. Die optimale Zuordnung der Studienplätze soll die durchschnittliche Zufriedenheit der Studierenden maximieren.

Gegeben ist die Klasse *StudentIn* mit der Methode

```
public double getZufriedenheit(Studienplatz stpl),
```

die die Zufriedenheit des Studierenden mit dem Studienplatz zurückliefert, und die Klasse *Studienplatz*. Implementieren Sie die Klasse *Studienverwaltung* mit der Methode

```
Map<StudentIn,Studienplatz> zuordnen(StudentIn[] studierende, Studienplatz[] plaetze),
```

die eine optimale Zuordnung der Studienplätze zu den Studierenden berechnet. (Sie können annehmen, dass *studierende.length == plaetze.length* ist.)

*Hinweis*: Es empfiehlt sich, die optimale Zuordnung durch vollständiges Ausprobieren aller Möglichkeiten zu berechnen.