

Name: _____

Matrikelnummer: _____

Aufgabe 1

Gegeben ist die Klasse *Auftrag* mit den Methoden

double getDauer(),

die die Bearbeitungsdauer für den Auftrag zurückliefert, und

Auftraggeber getAuftraggeber(),

die den Auftraggeber des Auftrags zurückliefert. Die Klasse *Auftraggeber* ist ebenfalls gegeben und enthält die Methode

void informiere(Auftrag einAuftrag, int schichtnummer).

Schreiben Sie die Klasse *Planung* mit der Methode

void plane(Auftrag[] auftragsliste, double schichtlaenge),

die für jeden Auftrag berechnet, in welcher Schicht er bearbeitet wird, und den Auftraggeber darüber mittels der Methode *informiere(Auftrag einAuftrag, int schichtnummer)* informiert. Bei der Berechnung der Schicht sind folgende Punkte zu beachten:

- Die Schichten sind fortlaufend mit 1 beginnend nummeriert, also 1,2,...
- Die Reihenfolge der Aufträge in der Auftragsliste darf nicht verändert werden.
- Die Bearbeitung eines Auftrags darf nicht unterbrochen werden und muss innerhalb einer Schicht fertig gestellt werden. (Sie können davon ausgehen, dass die Bearbeitungsdauer jedes Auftrags kleiner als die Schichtlänge ist.)
- Leerläufe in Schichten sind nur dann zulässig, wenn der nächste Auftrag in dieser Schicht nicht mehr fertig gestellt werden könnte.

Bespiel: Bei einer Schichtlänge von 10 und Bearbeitungsdauern 5,3,3,8,1 werden die ersten beiden Aufträge in der ersten Schicht, der dritte Auftrag in der zweiten Schicht, und der vierte und fünfte Auftrag in der dritten Schicht bearbeitet.

Aufgabe 2

Gegeben ist die Klasse *Transportgut* mit der Methode

double getGewicht().

Schreiben Sie eine Klasse *Auswahl* mit der Methode

Transportgut[] waehleMaxAnzahl(Transportgut[] transportliste, double maxGewicht),

die aus der *transportliste* möglichst viele Transportgüter zum Transport auswählt, ohne dass das Gewicht der ausgewählten Transportgüter das *maxGewicht* übersteigt. Die Liste der ausgewählten Transportgüter soll von der Methode zurückgeliefert werden. (Die Reihenfolge, in der die ausgewählten Transportgüter zurückgeliefert werden, ist beliebig.)

Beispiel: Bei einem Gewicht der Transportgüter 8,2,7,5,3,6 und einem *maxGewicht*=10 sollen die Transportgüter mit den Gewichten 2,5,3 ausgewählt werden.

Aufgabe 3

Gegeben ist das Interface *Folge*:

```
interface Folge
{
    /** Liefert das n-te Folgenglied zurück */
    public double get(int n);
}
```

Weiters ist die Klasse *EndlicheFolge* gegeben, die das Interface *Folge* implementiert und weiters über den Konstruktor

EndlicheFolge(Folge eineFolge, int laenge),

sowie die Methode

int getLaenge()

verfügt. Die Folgenglieder von *EndlicheFolge* sind mit 1,...,*laenge* nummeriert. Wird ein anderes Folgenglied abgefragt, wird ein beliebiger Wert zurückgeliefert.

Implementieren Sie die Unterklasse *AufsteigendeEndlicheFolge* von *EndlicheFolge* mit der Methode

static AufsteigendeEndlicheFolge create(EndlicheFolge eineFolge),

die eine *AufsteigendeEndlicheFolge* mit den Folgengliedern von *eineFolge* (in unveränderter Reihenfolge) zurückliefert, wenn diese Folgenglieder tatsächlich aufsteigend sortiert sind, und die *null* zurückliefert, wenn das nicht der Fall ist.