

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

Mat.Nr.: \_\_\_\_\_

## Aufgabe 1

Gegeben ist die Klasse *Taetigkeit*:

```
class Taetigkeit
{
    Taetigkeit(String name) {
        ...
    }

    /**
     * Liefert true wenn die Taetigkeit abgeschlossen ist.
     */
    boolean isErledigt() {
        ...
    }

    public String toString() {
        ...
    }

    ...
}
```

Schreiben Sie eine Unterklasse *Projekt* der Klasse *Taetigkeit* mit einem Konstruktor

```
Projekt(String name, ArrayList<Taetigkeit> teilaufgaben),
```

wobei in *teilaufgaben* jene Tätigkeiten enthalten sind, die für den Abschluss des Projekts notwendig sind.

Überschreiben Sie in der Klasse *Projekt* die Methode *isErledigt()* so, dass die Methode dann *true* liefert, wenn alle Teilaufgaben erledigt sind.

Überschreiben Sie auch die Methode *toString()*, sodass zu dem von *toString()* in *Taetigkeit* erzeugten String auch noch die Anzahl der Teilaufgaben angefügt wird.

## Aufgabe 2

Gegeben sind Arbeitsaufträge unterschiedlicher Dauer, die benötigte Anzahl von Stunden für jeden Auftrag ist in einem *int*-Array hinterlegt. Schreiben Sie eine Methode

```
int getMaxAnzahl(int [] auftragslaengen, int maxStunden),
```

die die maximale Anzahl der Aufträge zurückliefert, die in *maxStunden* erledigt werden können.

*Hinweis:* Diese maximale Anzahl  $n(0, \text{auftragslaengen}, \text{maxStunden})$  kann mit Hilfe der Rekursion

$$n(i, a, m) = \begin{cases} 0 & \text{wenn } i \geq a.\text{length} \\ n(i + 1, a, m) & \text{wenn } a[i] > m \\ \max\{1 + n(i + 1, a, m - a[i]), n(i + 1, a, m)\} & \text{sonst} \end{cases}$$

berechnet werden. Auch eine nicht-rekursive Lösung ist möglich.