

Informationen: Die Beispiele sind bis 14.11., 8 Uhr abzugeben bzw. bis zur Übung vorzubereiten.

Beispiel 6.1:

Überschreiben Sie in den Klassen *Zeitpunkt* (aus Beispiel 1.7 oder 2.1) und *Termin* (aus Beispiel 1.8 oder 2.2) die Methoden

```
public boolean equals(Object obj),  
public int hashCode().
```

Zwei Instanzen von *Zeitpunkt* sollen genau dann gleich sein, wenn alle Attribute gleich sind. Zwei Termine sollen dann gleich sein, wenn ihre Attribute (im Sinne von *equals*) gleich sind. Überlegen Sie sich einen passenden Hashcode für *Zeitpunkt*- und *Termin*-Objekte.

Beispiel 6.2:

Gegeben sei die Klasse *Schiff* mit Konstruktor

```
Schiff(double laenge, double geschwindigkeit)
```

und einer Methode

```
double getGeschwindigkeit(),
```

die die Geschwindigkeit des Schiffs zurückgibt. In der Klasse *Schiff* wird außerdem die Methode *toString()* aus *Object* so überschrieben, dass diese die Länge und die Geschwindigkeit des Schiffs in einem String der Form "Länge:laenge/Geschwindigkeit:geschwindigkeit" zurückgibt.

Schreiben Sie eine Unterklasse *Frachtschiff* der Klasse *Schiff* mit Konstruktor

```
Frachtschiff(double laenge, double geschwindigk, double ladevolumen)
```

und überschreiben Sie die Methode *toString()* so, dass in dem Rückgabestring auch das Ladevolumen zurückgegeben wird: "Länge:laenge/Geschwindigkeit:geschwindigkeit/Ladevolumen:ladevolumen".

Beispiel 6.3:

Gegeben sei die Klasse *Geschenk*, in der die Methoden *equals* und *hashCode* aus *Object* überschrieben sind. Schreiben Sie eine Klasse *Nikolosack* mit Konstruktor

```
public Nikolosack(),
```

der einen leeren *Nikolosack* erzeugt.

Die Methode

```
public void add(Geschenk einGeschenk)
```

soll *einGeschenk* in den Sack geben.

Die Methode

```
public HashSet<Geschenk> getAll()
```

soll alle Geschenke im Sack in einem *HashSet* zurückgeben.

Die Methode

```
public Geschenk getOne()
```

soll ein beliebiges Geschenk aus dem Sack entfernen und zurückgeben.

Beispiel 6.4:

Schreiben Sie Beispiel 4.7 so um, dass die Methode *wahleProdukte* als Rückgabewert und ersten Parameter ein *HashSet* statt einer *ArrayList* hat. Überlegen Sie sich insbesondere, welche Attribute ein Produkt haben sollte, und wie man die Methoden *equals* und *hashCode* aus *Object* in *Produkt* überschreiben könnte.

Beispiel 6.5: Gegeben sei die Klasse *Bahnhof*, die Klasse *Hauptbahnhof*, die von *Bahnhof* erbt, sowie die Klasse *Zug*. Die Klasse *Zug* verfügt über einen Konstruktor

```
public Zug(String nummer, Bahnhof start, Bahnhof ziel),
```

der einen Zug mit angegebener Nummer von *start* nach *ziel* anlegt. Weiters verfügt die Klasse *Zug* über eine Methode

```
public ArrayList<Bahnhof> gibStrecke(),
```

die alle Bahnhöfe auf der Strecke des Zuges in einer *ArrayList* zurückgibt.

Schreiben Sie eine Klasse *Eilzug* mit einem Konstruktor Ihrer Wahl, die von *Zug* erbt. Überschreiben Sie dabei die Methode *gibStrecke()* so, dass nur die Hauptbahnhöfe der Strecke in einer *ArrayList* zurückgegeben werden.

Beispiel 6.6: Gegeben sei eine Klasse *Telefon* mit Konstruktor

```
Telefon(String nummer),
```

der ein Telefon mit der angegebenen Nummer erstellt. Die Klasse *Telefon* verfügt außerdem über die folgenden Methoden:

```
public void anrufen(Telefon andresTelefon)
```

ruft *andresTelefon* an,

```
public String gibNummer()
```

gibt die Nummer des Telefons als String zurück.

Schreiben Sie eine Klasse *TelefonMitDisplay*, die von *Telefon* erbt. Überlegen Sie sich einen passenden Konstruktor, der ein Telefon mit leerem Display erstellt. Weiters soll es die folgenden Methoden geben:

```
public void setDisplay(String s)
```

soll den String *s* ins Display schreiben,

```
public void loeschen()
```

soll das Display löschen.

Überschreiben Sie in *TelefonMitDisplay* außerdem die Methode *anrufen* so, dass im aufrufenden Objekt der Anruf im Display angezeigt wird (z.B. String „Anruf an 0664/1234567“). Falls das angerufene Telefon ein Objekt der Klasse *TelefonMitDisplay* ist, soll im Display der Anruf ebenfalls verzeichnet werden (z.B. String „Anruf von 0664/1234567“).

Beispiel 6.7: [Triple.java]

Schreiben Sie eine Klasse *Triple* mit einer Methode

```
static boolean containsTriple(int[] liste),
```

die *true* zurückliefert, wenn es eine Zahl gibt, die in *liste* genau dreimal vorkommt.