

Name:

Matrikelnummer:

Bearbeitungszeit: 60 min.

Im Paket `musik.geg` sind gegeben die Klassen `Musiktitel` und `Person`, das Interface `MusikRanking`, und die Klasse `RankingPerson`, die das Interface `MusikRanking` implementiert.

Das Interface `MusikRanking` erfordert die Methode

```
public int compare(Musiktitel titel1, Musiktitel titel2),
```

die einen negativen Wert zurück geben soll, wenn `titel1` besser gefällt als `titel2`, einen positiven Wert, wenn `titel2` besser gefällt als `titel1`, und 0, wenn die beiden Titel gleich gut gefallen.

Die Klasse `RankingPerson` besitzt den Konstruktor

```
public RankingPerson(Person einePerson),
```

und die vom Interface `MusikRanking` geforderte Methode `compare()` ist so implementiert, dass sie dem Musikgeschmack der übergebenen Person entspricht.

Schreiben Sie die Klasse `musik.RankingPersonenliste` mit dem Konstruktor

```
public RankingPersonenliste(List<Person> personenliste),
```

die das Interface `MusikRanking` so implementiert, dass die Methode `compare(titel1,titel2)` einen negativen Wert liefert, wenn mehr Personen aus der `personenliste` den `titel1` gegenüber dem `titel2` bevorzugen, einen positiven Wert, wenn mehr Personen den `titel2` bevorzugen, und 0, wenn gleich viele Personen den `titel1` bzw. den `titel2` bevorzugen. (Personen, denen beide Titel gleich gut gefallen, werden jeweils nicht mitgezählt.)

Beispiel: In den folgenden Tabellen sind die Vorlieben von 4 Personen bzgl. dreier Musiktitel (A,B,C) als Ergebnis von `compare(tit1,tit2)` dargestellt:

tit1\tit2	A	B	C
A	0	-1	-1
B	+1	0	-1
C	+1	+1	0

tit1\tit2	A	B	C
A	0	-3	+2
B	+9	0	+3
C	-1	-2	0

tit1\tit2	A	B	C
A	0	+1	+1
B	-1	0	0
C	-1	0	0

tit1\tit2	A	B	C
A	0	0	-1
B	0	0	-2
C	+4	+3	0

Ein mögliches Ergebnis der Methode `compare(tit1,tit2)` von `RankingPersonenliste` für die Liste dieser Personen ist folgendes (nur die Vorzeichen müssen übereinstimmen):

tit1\tit2	A	B	C
A	0	-1	0
B	+1	0	-1
C	0	+1	0

Testen: Zum Testen Ihrer Lösung stellt die Klasse `Musiktitel` die Methoden

```
public static List<Person> getBeispielPersonen(),  
public static Musiktitel[] getBeispielTitel()
```

zur Verfügung, die die 4 Personen und die 3 Musiktitel des obigen Beispiels zurückliefern. Weitere Personen und Musiktitel können mit den parameterlosen Konstruktoren der Klassen erzeugt werden.

Die Methode

```
public static void printRanking(MusikRanking ranking, Musiktitel[] titel)
```

gibt das Ranking für die übergebenen Musiktitel wie in den obenstehenden Tabellen aus.