

## **Hausaufgaben 6**

**11.11.2021**

Abgabe der Lösung am 17.11.2021

### **Pokerspiel**

Im sogenannten Heads Up treten beim Pokerspiel zwei Spieler gegeneinander an, die jeweils 3 Karten (= ein Blatt) auf der Hand halten. Es gewinnt der Spieler, dessen Blatt eine höhere Wertigkeit als das des anderen hat. In dieser Aufgabe wird eine vereinfachte Variante dieses Spiels gespielt. Jede Karte hat einen Wert zwischen 2 (inklusive) und 14 (inklusive). Folgende Situationen können dabei, aufsteigend nach der Wertigkeit sortiert, auftreten:

- Paar - genau zwei Karten des Blattes haben denselben Wert.
  - Drilling - genau drei Karten des Blattes haben denselben Wert.
- 
- a) Implementieren Sie eine Klasse `Blatt`, deren einziges Attribut ein Integer-Feld der Länge 3 ist, dessen Einträge Werte von Karten symbolisieren.
  - b) Die Klasse soll einen Konstruktor haben, der ein `int`-Feld übergeben bekommt und das Attribut setzt. Achten Sie auf die Datenkapselung. Werfen Sie eine passende Exception, falls das übergebene Feld nicht korrekt ist.
  - c) Fügen Sie der Klasse `Blatt` eine `toString`-Methode hinzu, die die Werte der Karten als String zurückgibt.
  - d) Schreiben Sie eine zweite Klasse `BlattVergleich`, die das Java-Interface `Comparator<Blatt>` implementiert. Diese Klasse wird in der Testmethode (siehe Rückseite) benutzt, um zwei Pokerblätter miteinander zu vergleichen.

Die Vergleichskriterien sind wie folgt:

- Drilling schlägt Paar, Paar schlägt komplett unterschiedliche Karten.
- Bei 2 Drillingen bzw. Paaren zählt das mit dem höheren Wert.
- Bei 2 Paaren mit gleichem Wert zählt die 3. Karte.
- Bei komplett unterschiedlichen Karten zählt die Summe der drei Karten.

Testfall

```

public static void main(String[] args) {
    //Verschiedene Blaetter zum Testen erzeugen
    Blatt drillingNiedrig = new Blatt(new int [] { 2,2, 2});
    Blatt drillingHoch   = new Blatt(new int [] { 9,9, 9});

    Blatt paarNiedrig   = new Blatt(new int [] { 2,2, 3});
    Blatt paarHoch     = new Blatt(new int [] { 2,5, 5});
    Blatt paarHoch2    = new Blatt(new int [] {10,5, 5});

    //Weder Paar noch Drilling, alle Karten sind unterschiedlich
    Blatt einfachNiedrig = new Blatt(new int [] { 2,3, 8});
    Blatt einfachHoch   = new Blatt(new int [] { 3,4,10});

    //Verschiedene Blaetter testen
    BlattVergleich bv = new BlattVergleich();

    werteAus(paarHoch,      drillingNiedrig, bv);
    werteAus(paarNiedrig,  einfachNiedrig,  bv);
    werteAus(drillingHoch, drillingNiedrig, bv);
    werteAus(paarHoch,    paarNiedrig,     bv);
    werteAus(paarHoch,    paarHoch2,      bv);
    werteAus(einfachNiedrig, einfachHoch,   bv);
}

/**
 * Die Methode prueft fuer zwei Blaetter, welches gewonnen hat, und gibt das
 * Ergebnis aus.
 * @param b1 Blatt 1
 * @param b2 Blatt 2
 * @param cb Comparator, der zum Vergleich der Blaetter verwendet wird
 */
private static void werteAus(Blatt b1, Blatt b2, Comparator<Blatt> cb){
    int vergleich = cb.compare(b1, b2);
    String ergebnis = "";
    if(vergleich < 0){
        ergebnis = " verliert gegen ";
    }else if(vergleich == 0){
        ergebnis = " ist gleichwertig mit ";
    }else{
        ergebnis = " schlaegt ";
    }

    System.out.println(b1 + ergebnis + b2);
}

```

Ausgabe

```

2, 5, 5 verliert gegen 2, 2, 2
2, 2, 3 schlaegt 2, 3, 8
9, 9, 9 schlaegt 2, 2, 2
2, 5, 5 schlaegt 2, 2, 3
2, 5, 5 verliert gegen 10, 5, 5
2, 3, 8 verliert gegen 3, 4, 10

```