

# Programmieren und Software-Engineering II

## Übung 8

Name: \_\_\_\_\_ Klasse: \_\_\_\_\_ Datum: \_\_\_\_\_

### Aufgabe 1: Bruch

Es soll ein Java-Programm implementiert werden, das eine Klasse beinhaltet, die das Rechnen mit Brüchen realisiert. Dabei soll die Klasse Operationen für das Addieren, Subtrahieren, Multiplizieren und Dividieren enthalten. Die Ergebnisse sollen wiederum in eigenen Objekten derselben Klasse zurückgegeben werden. Weiters sollen die Ergebnisse nur Brüche mit kleinstmöglichem Nenner enthalten, d.h. es muss entsprechend gekürzt werden. Um Brüche zu addieren / subtrahieren, geht man folgendermaßen vor:

Zuerst macht man sie gleichnamig, das heißt, man bringt sie auf den gleichen Nenner. Das erreicht man am einfachsten, indem man den ersten Bruch mit dem Nenner des zweiten Bruches erweitert und den zweiten Bruch mit dem Nenner des ersten.

Beispiel:  $\frac{3}{5} + \frac{7}{3} = \frac{9}{15} + \frac{35}{15} = \frac{44}{15}$

Hier sieht man schön, da\_ der erste Bruch mit 3, also dem Nenner des zweiten erweitert wurde, und der zweite Bruch mit dem Zähler des ersten. Anschließend addiert bzw. subtrahiert man einfach die Zähler der beiden Brüche. Das Ergebnis sollte man noch kürzen, d.h. mit dem größten gemeinsamen Teiler dividieren.

Beim Multiplizieren werden jeweils die beiden Zähler und die beiden Nenner miteinander multipliziert und das Ergebnis anschließend gekürzt.

Beispiel:  $\frac{3}{4} \cdot \frac{6}{8} = \frac{18}{32} = \frac{9}{16}$

Das Dividieren von Brüchen kann auf das Multiplizieren mit dem Kehrwert des zweiten Bruchs zurückgeführt werden:

Beispiel:

$$\frac{\frac{5}{8}}{\frac{3}{2}} = \frac{5}{8} \cdot \frac{2}{3} = \frac{10}{24} = \frac{5}{12}$$

Ein Testbeispiel in dem die Klasse verwendet wird:

```
public static void main(String[] args) {
    Fraction a = new Fraction(3,5);
    Fraction b = new Fraction(7,3);
    System.out.println(a.toString()+"+"+b.toString()+"="+a.add(b).toString());
    a.setFraction(3, 4);
    b.setFraction(3, 5);
    System.out.println(a.toString()+"-"+b.toString()+"="+a.sub(b).toString());
    a.setFraction(3, 4);
    b.setFraction(6, 8);
    System.out.println(a.toString()+"*"+b.toString()+"="+a.mul(b).toString());
    a.setFraction(5, 8);
    b.setFraction(3, 2);
    System.out.println(a.toString()+"/"+b.toString()+"="+a.div(b).toString());
}
```

Die Ausgabe des Testprogrammes:

```
3/5+7/3=44/15
3/4-3/5=3/20
3/4*6/8=9/16
5/8/3/2=5/12
```

# Programmieren und Software-Engineering II

## Übung 8

### Aufgabe 2: Artikel einscannen und Rechnung drucken

Implementiere ein vollständiges Java-Programm für die Simulation eines Einkaufs an einer Supermarkt-Kassa, wo einzelne Artikel gescannt werden und anschließend eine Rechnung gedruckt wird.

Folgende Funktionalitäten sollen verfügbar sein:

- Artikel scannen und zur Rechnung hinzufügen
- Die Rechnung kann beliebig viele Artikel enthalten
- Der zuletzt gescannte Artikel kann storniert werden
- Berechnen der Gesamtsumme einer Rechnung
- Einfache Ausgabe der Rechnung (alle Artikel und Gesamtpreis)
- Sortieren der Rechnung nach dem Preis
- Ausgabe einer kompakten Liste, d.h. Artikel mit demselben Namen (Methode equals...) werden summiert

Das **Hauptprogramm** besitzt folgenden **Aufbau**:

```
public static void main(String[] args){
    Article a1 = new Article ();
    Article a2 = new Article ("Banane"); // Preis wird auf 1.0 gesetzt
    Article a3 = new Article ("Apfel", 0.60);

    Invoice inv1 = new Invoice (1); // Rechnung mit der Rechnungsnummer 1
    inv1.addArticle(new Article ("Apfel", 0.70));
    inv1.addArticle (new Article ("Apfel", 0.50));
    inv1.cancelArticle(); // letzter Artikel wird wieder gelöscht
    inv1.addArticle (new Article ("Käse", 2.70));
    inv1.addArticle (new Article ("Apfel", 0.80));
    inv1.addArticle (new Article ("Apfel", 0.90));
    inv1.addArticle (new Article ("Käse", 3.50));
    inv1.addArticle (new Article ("Birne", 1.20));
    inv1.addArticle (new Article ("Orange", 0.90));
    inv1.addArticle (new Article ("Birne", 0.70));
    inv1.addArticle (new Article ("Kiwi", 0.40));
    inv1.addArticle (new Article ("Kiwi", 0.50));
    inv1.addArticle (new Article ("Birne", 0.70));

    System.out.println(inv1.toString()); // einfache Ausgabe

    inv1.printKompact(); // Kompakte Ausgabe

    inv1.sortPrice();
    System.out.println(inv1.toString()); // Sortieren und einfache Ausgabe
```