



## Übung 8

### Aufgabe 1: Brüche

In der folgenden Aufgabe geht es um Funktionen, Prozeduren und Datenstrukturen. Es soll eine kleine Bibliothek zum Bruchrechnen implementiert werden.

Zum Erstellen einer solchen Bibliothek gehört der Entwurf einer Datenstruktur, die einen Bruch repräsentiert, sowie der Entwurf von Prozeduren und Funktionen, welche Operationen auf diese Datenstruktur anbieten.

Als zu implementierende Prozeduren und Funktionen haben wir uns folgende Operationen überlegt:

**Initialisieren** eine Prozedur oder Funktion, mit der man Zähler und Nenner eines Bruches festlegen kann.

*Eingabe:* Zähler und Nenner.

*Ausgabe:* der durch diese beiden Werte festgelegte Bruch.

**Kopieren** eine Prozedur oder Funktion, mit der man einen Bruch in einen anderen Bruch kopieren kann.

*Eingabe:* ein Bruch.

*Ausgabe:* Kopie des eingegebenen Bruches.

**Erweitern** eine Prozedur oder Funktion, mit der man einen Bruch erweitern kann.

*Eingabe:* ein Bruch und eine Zahl.

*Ausgabe:* der um die Zahl erweiterte Bruch.

**Kürzen** eine Prozedur oder Funktion, mit der man einen Bruch kürzen kann.

*Eingabe:* ein Bruch.

*Ausgabe:* der gekürzte Bruch.

**Addition** eine Prozedur oder Funktion, mit der man zwei Brüche addieren kann.

*Eingabe:* zwei Brüche.

*Ausgabe:* Summe der beiden Brüche.

**Subtraktion** eine Prozedur oder Funktion, mit der man zwei Brüche subtrahieren kann.

*Eingabe:* zwei Brüche.

*Ausgabe:* Differenz der beiden Brüche.

**Multiplikation** eine Prozedur oder Funktion, mit der man zwei Brüche multiplizieren kann.

*Eingabe:* zwei Brüche.

*Ausgabe:* Produkt der beiden Brüche.

**Division** eine Prozedur oder Funktion, mit der man zwei Brüche dividieren kann.

*Eingabe:* ein Dividend-Bruch und ein Divisor-Bruch.

*Ausgabe:* Quotient der beiden Brüche.

**Auswertung** eine Prozedur oder Funktion, mit der man einen Bruch in eine reelle Zahl umwandeln kann.

*Eingabe:* ein Bruch.

*Ausgabe:* reelle Zahl, die den Bruch repräsentiert.

Die Idee bei der Implementation solcher Operationen ist, dass der Programmierer gar nicht genau weiss, wie die Datenstruktur intern aufgebaut ist, wenn er sie benutzt. Er soll allein durch die Operationen mit der Datenstruktur arbeiten können.

- a) Entwerfen Sie eine geeignete Datenstruktur um Brüche darzustellen.
- b) Entwerfen Sie für jede der oben besprochenen Operationen eine Funktion oder Prozedur. Hauptaugenmerk ist bei dieser Aufgabe wiederum der Entwurf der Prozedur- und Funktionsschnittstelle. Beachten Sie dabei, dass man in mehreren unterschiedlichen formalen Parametern gleiche aktuelle Parameter übergeben kann.

**Beispiel:** Für die Additionsoperation soll folgendes möglich sein:  $a = a + a$ .

- c) Folgendes Programmstück wird hier als Pseudonotation angegeben,

$m = \frac{1}{2} \quad t := \frac{1}{1} \quad v = t$   
für alle  $i \in \mathbb{Z}$  von 0 bis 14

$t = m \cdot t \quad v = v - t$

$t = m \cdot t \quad v = v + t$

BildschirmAusgabe: reeler Wert von  $v$

berechnet:

$$1 - \frac{1}{2} + \frac{1}{4} - \frac{1}{8} + \dots = \sum_{n=0}^{31} \frac{(-1)^n}{2^n} = \frac{715827883}{1073741824} \approx \frac{2}{3}$$

Testen Sie Ihre Bibliothek mit diesem Programmstück als Hauptprogramm. Das Programm erzeugt folgende Ausgabe, anhand derer Sie Ihre Version testen können:

0.75  
0.6875  
0.671875  
0.667969  
0.666992  
0.666748  
0.666687  
0.666672  
0.666668  
0.666667  
...