

Versuchen Sie immer, Ihren Code zu kommentieren!

Erstellen Sie zuerst ein Klassendiagramm für die Aufgaben!

Aufgabe 1

Schreiben Sie eine Klasse Bruch, die zwei Attribute Zähler und Nenner vom Typ int enthält. Schreiben Sie zwei Konstruktoren für die Klasse Bruch: ein Konstruktor mit einem Parameter für den Zähler (der Nenner soll mit dem Wert 1 initialisiert werden) und einen Konstruktor mit zwei Parametern für die beiden Attribute.

Schreiben Sie die folgenden Methoden:

- Eine Funktion Ausgabe(), die einen Bruch auf der Standardausgabe ausgibt (z.B. 1/4, 12/20, etc.)
- Eine Funktion GetKehrwert(), die einen Bruch umkehrt, d.h. Zähler wird zu Nenner und Nenner wird zum Zähler (Kehrwert bilden).
- Eine Funktion Kürzen(), die einen Bruch kürzt, d.h. Zähler und Nenner durch den größten gemeinsamen Teiler (ggT) dividiert.
- Die Funktionen Addiere(), Subtrahiere(), Multipliziere() und Dividiere() für die vier Grundrechenarten (+, -, * und /). Die Funktionen erhalten als Argument einen Bruch und liefern das gekürzte Ergebnis der Rechenoperation als Bruch mittels return zurück.

Testen Sie das Ergebnis anhand einiger geeigneter Instanziierungen und Berechnungen in der Main() Methode.

Aufgabe 2

In den 60er Jahren wurde eine Methode der Überprüfung von Identifikationsnummern entwickelt. Es handelt sich um eine simple Prüfsummen-Formel, die benutzt wird, um eine Vielzahl von Identifikationsnummern wie z.B. Kreditkartennummern zu bestätigen. Der Algorithmus wird als einfache Methode benutzt, um gültige Nummern von zufälligen Nummernkombinationen zu unterscheiden. Die Formel generiert eine Prüfziffer, welche in der Regel an die unvollständige Identifikationsnummer angehängt wird, um die vollständige Nummer zu erhalten. Der Ablauf lautet wie folgt:

1) Beginnend mit der zweitletzten Ziffer und nach links durchlaufend, verdopple den Wert jeder zweiten Ziffer. Für jede Ziffer, aus welcher 10 oder mehr wird, bilde die Quersumme (heißt: addiere die einzelnen Ziffern). Aus den Ziffern 0,1,2,3,4,5,6,7,8,9 werden also der Reihe nach die Ziffern 0,1,4,3,8,5,3,7,7,9. Damit irrtümliche Vertauschungen der Ziffernreihenfolge die Prüfziffer verändern, wird diese Operation nur auf jede zweite Ziffer angewendet. Insgesamt wird zum Beispiel 1111 zu 2121 oder 8763 wird zu 7733 (von 8 verdoppeln ist 16 -> (1+6)=7 und 6 verdoppeln ist 12 -> (1+2)=3).

2) Addiere all diese Zahlen zusammen. Beispiel: 1111 wird 2121, dann $2+1+2+1=6$; 8763 wird 7733, dann $7+7+3+3=20$.

3) Wenn die Gesamtsumme Modulo 10 gleich 0 ist, dann ist die Nummer nach dem Algorithmus gültig, sonst nicht. Also ist 1111 nicht gültig (siehe oben, die letzte Ziffer der Gesamtsumme ist 6), 8763 aber schon (siehe oben, die letzte Ziffer von 20 ist 0).

Beispielidentifikationsnummer: 446-667-651

Ziffer	Verdoppelt	Reduziert	Summe der Ziffern
1		1	1
5	10	1+0	1
6		6	6
7	14	1+4	5
6		6	6
6	12	1+2	3
6		6	6
4	8	8+0	8
4		4	4

Gesamtsumme: 40

Schreiben Sie eine Klasse Prüfsumme mit der Klassen-Methode CheckIdentNumber(), welcher ein Array mit einer Nummer übergebenen wird. Ist die Nummer in Ordnung, liefert die Methode true zurück ansonsten false.

Zusatz: Legen Sie eine überladene Methode an, welcher Sie einen String mit einer Prüfnummer übergeben können.

Aufgabe 3

Gegeben ist eine Datei „Personen.txt“ mit folgendem Inhalt (selbst erstellen!!):

Tim:1.12.2001
Tom:1.12.1991
Eva:4.3.1997
Tina:5.2.1987
Sven:23.06.1982
Sabine:14.09.1981

Mit Hilfe der Inhalte aus der Datei soll Ihr Programm folgende Ausgaben machen:

- Es soll zu jeder Person in der Datei das Alter ermittelt werden (in Jahren).
- Es soll das Durchschnittsalter der Gruppe ermittelt werden.

Erstellen Sie eine Klasse **Person** mit den Attributen **Name** und **Geburtsdatum** und den nötigen Methoden. Verwenden Sie für das Geburtsdatum den Datentyp **DateTime**. Mittels **DateTime.Today** bekommen Sie das aktuelle Datum heraus. Verwenden Sie **DateTime.Parse**

um einen String in ein DateTime umzuwandeln. Legen Sie ein Array vom Typ Person an, in dem Sie alle Personen speichern. Erzeugen Sie pro Zeile der Datei ein Personen-Objekt.

Zusatz: Stellen Sie fest, ob die Person in diesem Jahr schon Geburtstag hatte und passen Sie das Alter ggf. an.