

3. Übung

Abgabe: bis 22.11.2021, 6 Uhr

Abnahme DFHI: 22.11.2021

Abnahme PI: 23.11.2021

Entwickeln sie eine Klasse MathFunctions, die ähnlich wie die Klasse java.lang.Math Funktionen in Form von Klassenmethoden anbietet und eine Dialogklasse zum interaktiven Testen dieser Klasse. Vorläufig anzubietende Funktionen:

- *public static long berechneTeilersumme (long zahl):* Teilersumme zu einer übergebenen natürlichen Zahl berechnen und zurückgeben.
- *public static String berechneChecksummeIsbn(long isbn):* Prüfziffer für eine ISBN-10 berechnen
- *public static String berechneNullstellen (double p, double q):* Nullstellen einer quadratischen Gleichung bestimmen und als aufbereiteten String zurückgeben.

Erstellen Sie eine Testdokumentation mit sinnvollen Testwerten, an denen die Funktionstüchtigkeit Ihrer Lösung überprüft werden kann. Achten Sie darauf, dass ein einziger Testwert pro Methode nicht ausreicht, um deren Funktionstüchtigkeit zu beweisen. Orientieren Sie sich an der Beispiel-Testdokumentation, die Sie im Moodle finden können. Laden Sie Ihre Testdokumentation als pdf-Datei in das Abgabeverzeichnis hoch und bringen Sie einen Ausdruck zur Abnahme mit.

Aufgabe 1

Die Methode berechneTeilersumme(long zahl) berechnet die Teilersumme einer natürlichen Zahl.

Beispiel: $\text{teilersumme}(6) = 1 + 2 + 3 + 6 = 12$

Aufgabe 2

Die Internationale Standardbuchnummer, abgekürzt ISBN, ist eine Nummer zur eindeutigen Kennzeichnung von Büchern und anderen Veröffentlichungen mit redaktionellem Anteil. Es gibt die ältere ISBN-10, bestehend aus 10 Ziffern und die neuere ISBN-13, bestehend aus genau 13 Ziffern. Die letzte Ziffer ist dabei eine sogenannte Prüfziffer, die aus den ersten 9 bzw. 12 Ziffern berechnet werden kann. Wir betrachten im Folgenden nur die ISBN-10.

Die Prüfziffer zu einer ISBN-10 wird wie folgt berechnet: Bezeichnet man die ersten neun Ziffern mit z_1 bis z_9 , so gilt für die Prüfziffer an der zehnten Stelle:

$$z_{10} = \left(\sum_{i=1}^9 \right) \bmod 11 = (1 * z_1 + 2 * z_2 + \dots + 9 * z_9) \bmod 11$$

Ergibt sich als Ergebnis der Wert 10, so wird als Prüfziffer das Zeichen X verwendet.

Beispiele:

386680192: $z_{10} = 1 * 3 + 2 * 8 + 3 * 6 + 4 * 6 + 5 * 8 + 6 * 0 + 7 * 1 + 8 * 9 + 9 * 2 = 198 \bmod 11 = 0$

383622862: $z_{10} = 1 * 3 + 2 * 8 + 3 * 3 + 4 * 6 + 5 * 2 + 6 * 2 + 7 * 8 + 8 * 6 + 9 * 2 = 196 \bmod 11 = 9$

Übergeben Sie der Methode die zu prüfende Zahl als long-Zahl und geben Sie die errechnete Prüfziffer als String zurück.

Tipp: Schauen Sie sich dazu nochmal das Vorlesungsbeispiel zum Errechnen der Quersumme einer natürlichen Zahl an.

Aufgabe 3

Eine quadratische Gleichung der Form

$$x^2 + px + q = 0$$

wird mit Hilfe der bekannten Formel

$$x_{1/2} = -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q} \text{ für } \left(\frac{p}{2}\right)^2 - q \geq 0$$

gelöst. Dabei sind die Fälle zu unterscheiden, ob zwei reelle, zwei komplexe oder doppelte Nullstellen vorliegen. Die komplexen Lösungen brauchen nicht ausgegeben zu werden.

Die Fallunterscheidung erfolgt auf Grund des Wertes der Diskriminante

$$D = \left(\frac{p}{2}\right)^2 - q$$

$$D = \begin{cases} > 0 & \text{zwei reelle Nullstellen} \\ = 0 & \text{eine doppelte Nullstelle} \\ < 0 & \text{zwei komplexe Nullstelle} \end{cases}$$

Das Ergebnis der Berechnung ist als String zurückzugeben. Dies sieht in den drei oben beschriebenen Fällen folgendermaßen aus:

1. "Zwei Nullstellen: " + x1 + "|" + x2
2. "Doppelte Nullstelle: " + x1
3. "Komplexe Nullstellen"

Beispielausgaben

p	q	Rückgabestring
2.5	1.5	"Zwei Nullstellen: -1.0 -1.5"
3.0	2.25	"Doppelte Nullstelle: -1.5"
1.5	2.5	"Komplexe Nullstellen"

Beachten Sie, dass bei der Gleitpunktarithmetik D nicht genau 0.0 wird, sondern nur ungefähr!

Bemerkungen

Die vorgegebenen Namen und Signaturen und Rückgabewerte für die Methoden sind unbedingt einzuhalten, damit Ihre Lösung automatisch auf Korrektheit überprüft werden kann!