

Modellierung und Programmierung 1 – Übungsserie 2

Abgabetermin: 22.11.2022, 23:00 Uhr
Abgabeformat: zip für die Programme
Max. Punkte: 42

Arrays, Kontrollstrukturen

1. Implementierung (42 Punkte)

a) Schreiben Sie eine Methode `createIntArray` mit folgenden Eigenschaften

- Parameter:
 - eine ganze Zahl `minimum`
 - eine ganze Zahl `maximum`
- Rückgabe: ein Array vom Typ `int`
- Funktionalität
 - Erzeugen eines Arrays vom Typ `int`
 - Füllen des Arrays mit den Werten von `minimum` bis `maximum` einschließlich der Grenzen
 - Rückgabe des erzeugten Arrays
- Bemerkung: Das Array sollte exakt so viel Platz verwenden, wie für die Speicherung der Werte notwendig ist.

b) Schreiben Sie eine Methode `arrayInhaltAusgeben` mit folgenden Eigenschaften

- Parameter: ein Array vom Typ `int`
- Rückgabe: keine
- Funktionalität
 - Ausgabe aller Elemente des Arrays
 - * in der Reihenfolge ihrer Indizes
 - * in einer Zeile
 - * durch Komma getrennt

c) Schreiben Sie eine Methode `quadriere` mit folgenden Eigenschaften

- Parameter: ein Array vom Typ `int`
- Rückgabe: ein neues Array vom Typ `int`
- Funktionalität
 - Erzeugung eines neuen Arrays
 - Quadrieren aller Elemente des alten Arrays und speichern der Quadrate im neuen Array
 - Beispiel (1, 2, 3, 4) -> (1, 4, 9, 16)
 - Rückgabe des neuen Arrays

d) Schreiben Sie eine Methode `testAufQuadratzahl` mit folgenden Eigenschaften

- Parameter: eine ganze Zahl *zahl*
- Rückgabe: `boolean`
- Funktionalität
 - Testen Sie *zahl* darauf, ob sie das Quadrat einer natürlichen Zahl *n* ist:

$$zahl = n^2, n \in \mathbb{N}$$

- Falls ja, gebe `true` zurück
- Falls nein, gebe `false` zurück
- Hinweis: Verwenden Sie `Math.sqrt` und `Math.floor`

e) Schreiben Sie eine Methode `pythagoraeischeZahlentripel` mit folgenden Eigenschaften

- Parameter:
 - eine ganze Zahl `minimum`
 - eine ganze Zahl `maximum`
- Rückgabe: Anzahl der gefundenen Zahlentripel
- Funktionalität
 - Erzeugen Sie mit bisher programmierten Methoden ein Array mit Quadratzahlen von `minimum` bis `maximum` einschließlich der Grenzen
 - Geben Sie das Ergebnis unter Verwendung der zuvor beschriebenen Methode aus
 - Berechnen Sie alle Differenzen zweier Elemente des Arrays von Quadratzahlen *array*:

$$differenz = array[j] - array[i], j > i$$

- Falls $\sqrt{differenz} \in \mathbb{N}$ und $differenz > array[i]$, geben Sie das Ergebnis gemäß folgendem Beispiel aus
 - * Inhalt des Arrays an der Position *i*: 9
 - * Inhalt des Arrays an der Position *j*: 25
 - * $differenz = 16$
 - * Ausgabe:

$$3^2 + 4^2 = 5^2$$
- Falls $\sqrt{differenz} \in \mathbb{N}$ und $differenz > array[i]$, erhöhen Sie die Anzahl der Pythagoräischen Zahlentripel um 1
- Geben Sie die Anzahl der Pythagoräischen Zahlentripel zurück

f) Testen Sie ihre Methoden

- Fügen Sie alle bisherigen Methoden der Klasse `PythagoraeischeZahlentripel` hinzu
- Fügen Sie zu allen Methoden den Modifikator `static` hinzu
- Erzeugen Sie die Main-Methode
- Führen Sie in der Main-Methode die Methode `pythagoraeischeZahlentripel` aus
- Geben Sie anschließend in der Main-Methode die Anzahl der gefundenen Pythagoräischen Zahlentripel aus (Zahl plus Bedeutung!).
- Verwenden Sie als Testwerte mindestens
 - $minimum = 1, maximum = 1000$
 - $minimum = 1001, maximum = 2000$

-
- Speichern Sie die Ausgabe in einem Text-File und fügen Sie dieses Text-File dem zip-File der Abgabe hinzu

Allgemeine Hinweise:

- Verwenden Sie ausschließlich die in den Aufgabenstellungen angegebenen Klassen und deren Methoden aus Bibliotheken
- Achten Sie auf angemessene Sichtbarkeiten.
- Achten Sie auf angemessene Benennungen der Parameter und Attribute (Verwenden Sie einen anderen Namen als das oben verwendete **array**!)