

Datum: 03.12.2018

Abgabe bis: 10.12.2018 11:30

Realisieren Sie die nachfolgenden Aufgaben unter Verwendung des Methoden- bzw. Prozedurkonzepts (siehe dazu 03-Methodenkonzept).

Hinweis: Erfinden Sie das Rad nicht neu. Nutzen Sie, wenn Sie es möchten, ihre Lösungen der Vorstunden wenn zur Realisierung.

Bevor Sie eine Aufgabe beginnen...

Erstellen Sie zunächst eine Klasse mit dem Namen `IhrName_uebung`.

Innerhalb dieser Klasse lösen Sie **zwei** der nachfolgenden Aufgaben.

Am Ende...

Ergänzen Sie ihr Hauptprogramm (also die `public static void main(String[] args)`-Methode) um eine entsprechende Benutzereingabe, die abhängig vom eingegeben Wert eine ihrer Methoden/Prozeduren ausführt.

Hinweis: Sollte die aufgerufene Methode Parameter erwarten, lassen Sie diese durch den Benutzer eingeben.

Erste Primzahl

Übernehmen Sie die folgende Rohform der Methode `produkt` in die Klasse und ergänzen die Auslassung (...) so, dass die gewünschte Funktionalität gegeben ist.

```
public static int allePrim(int zahl1, int zahl2){  
    // Die Methode gibt die erste Primzahlen zwischen zahl1 und zahl2 als Ergebnis  
    zurück.  
    ...}
```

Alle Rechtwinkligen

Programmieren Sie eine `public static`-Prozedur mit dem Namen `alleRechtwinkligen`. Die Prozedur besitzt keine Parameter.

Die Ausgabe der Prozedur sind alle Ganzzahl-Tripel (a,b,c) für die folgende Bedingungen erfüllt sind:

- Jeder der drei Ganzzahlen a , b und c liegt zwischen 1 und 100 (einschliesslich).
- a ist größer oder gleich b und b ist größer oder gleich c .

- Die drei Ganzzahlen repräsentieren die Seiten eines **rechtwinkligen Dreiecks**, d.h. es gilt der 'Satz des Pythagoras', nämlich: $a^2 = b^2 + c^2$.

Die Ausgabe des Tripels soll Zeilenweise folgende Form haben:

Nr. 19: 50, 40, 30

Größter gemeinsamer Teiler

Entwerfen Sie eine **public static**-Methode mit dem Rückgabotyp **int** namens ggT. Die Methode erhält zwei **int**-Parameter mit den Bezeichnungen zahlA und zahlB.

Die Methode liefert den *größten gemeinsamen Teiler* der Variablenwerte von zahlA und zahlB zurück.

Ein vollständig gekürzter Bruch

Entwerfen Sie eine **public static**-Prozedur mit der Bezeichnung Bruch. Die Prozedur erhält zwei **int**-Parameter mit den Namen zaehler und nenner.

Die Prozedur soll zunächst den *größten gemeinsamen Teiler* der beiden Zahlen bestimmen und diesen Verwenden, um den **vollständig gekürzten Bruch** auf der Konsole auszugeben.

Prim-Pärchen

als Prim-Pärchen oder auch *Prim-Doublette* bezeichnet werden zwei Primzahlen bezeichnet, deren Differenz gleich 2 ist. z.B. 3 und 5, 9 und 11 oder 1019 und 1021.

Programmieren Sie eine **public static int**-Methode namens primPaar. Die Methode erhält einen **int**-Parameter mit der Bezeichnung min.

primPaar sucht von min aufwärts nach solchen Prim-Pärchen.

Wird eines gefunden, soll die kleinere der beiden Primzahlen zurückgegeben werden.

Wird zwischen min und der größten Zahl vom Typ **int** (`Integer.MAX_VALUE`) kein Pärchen gefunden, liefert primPaar den Wert 0 als Ergebnis.