



Übungsblatt 3: Schleifen; Arrays

Ausgabe: 16.10.17

Abgabe: 23.10.17 12:00 Uhr elektronisch mittels Git

Aufgabe 1

Erstellen Sie ein Java-Programm (Klasse “Dividers”), das alle Teiler einer *natürlichen Zahl* berechnet. Der Benutzer soll zu Beginn aufgefordert werden, eine ganze Zahl einzugeben. Nachdem er seine Zahl eingetippt hat, soll das Programm alle Teiler dieser Zahl in einer Zeile ausgeben. Die Zahlen sollen jeweils durch ein Leerzeichen voneinander getrennt werden. Falsche Eingaben sollen durch eine Fehlermeldung “Eingabe ungültig” signalisiert werden. Das Programm endet nach Ausgabe der Teiler bzw. der Fehlermeldung.

Sie können sich darauf verlassen, dass nur ganze Zahlen eingegeben werden.

Wählen Sie vernünftige Namen für Ihre Variablen!

Aufgabe 2

Erstellen Sie ein Java-Programm (Klasse “DividersArray”), das alle Teiler einer *natürlichen Zahl* berechnet. Der Benutzer soll zu Beginn aufgefordert werden, eine ganze Zahl einzugeben. Nachdem er seine Zahl eingetippt hat, soll das Programm alle Teiler dieser Zahl in ein Array von ganzen Zahlen der Länge 35 schreiben. Dieses Array wird wie folgt vereinbart:

```
int[] dividers = new int[35];
```

Die erste gültige Indexposition ist die Position 0, mit `dividers[0]` wird darauf zugegriffen.

Nachdem alle Teiler in das Array geschrieben wurden, soll der (komplette) Inhalt des Arrays in einer Zeile ausgegeben werden. Dabei werden die enthaltenen Zahlen jeweils durch ein Leerzeichen getrennt.

Falsche Eingaben sollen durch eine Fehlermeldung “Eingabe ungültig” signalisiert werden. Das Programm endet nach Ausgabe der Teiler bzw. der Fehlermeldung.

Sie können sich darauf verlassen, dass nur ganze Zahlen eingegeben werden.

Wählen Sie vernünftige Namen für Ihre Variablen!

Aufgabe 3

Palindrome sind allgemein Worte oder Sätze, die vorwärts und rückwärts gelesen denselben Sinn ergeben. Für diese Aufgabe schränken wir Palindrome auf ganze Zahlen, d.h. auf Ziffernfolgen ein.

Beispiele sind die Zahlen 1234321 oder 1221. Aber auch einzelne Ziffern sind per Definition Palindrome.

Ihre Aufgabe besteht nun darin, ein Java-Programm (Klasse “*PalindromeNumber*”) zu erstellen, das für eine beliebige ganze Zahl zwischen 0 und 2 Milliarden (jeweils inklusive) prüft, ob sie ein Palindrom darstellt. Die Zahl wird mittels `readInt()` eingelesen. Ihr Programm soll nach der Prüfung entweder das Ergebnis “Palindrom erkannt” oder “kein Palindrom” oder “ungültige Eingabe” ausgeben.

Arbeiten Sie nur mit Zahlen vom Typ `int`, d.h. es sind keine anderen Zahl-Typen und keine `Strings` erlaubt!

Ein möglicher Algorithmus (Sie dürfen gerne anders vorgehen!):

- Zerlegen Sie die eingegebene Zahl in ihre Ziffern und schreiben Sie diese in ein `int`-Array.
- Vergleichen Sie jeweils die erste und die letzte dieser Ziffern.
- Merken Sie sich, ob mindestens eins dieser Ziffernpaare ungleich ist.
- Geben Sie das Ergebnis aus.

Allgemeines

- Die Aufgaben sind in Eclipse zu bearbeiten und beim Testat vorzuführen.
- Legen Sie für die Bearbeitung dieses Übungsblattes ein Paket namens `pr1.uebung03` an, in dem Sie Ihre Klassen anlegen.
- Erlaubt sind `MakeItSimple`-Funktionen (keine nicht besprochene Funktionalität aus der Java-Standard-Bibliothek) und das bisher erworbene Wissen aus den PR1-Vorlesungen. Sie müssen alle(!) von Ihnen verwendeten Konstrukte der Sprache sowie alle verwendeten Methoden, die nicht aus der Hilfsbibliothek `MakeItSimple` stammen, gut erklären und nötigenfalls im Testat selbst programmieren können!
- Sie geben ab, indem Sie vor Ende der Abgabefrist Ihr Projekt mit den lauffähigen Programmen des Übungsblattes in das Repository auf *ovid* pushen. Achten Sie darauf, ob Sie die Kontroll-E-Mail bekommen! Die letzte hochgeladene Version Ihres Projekts wird gewertet. Andere Abgaben, ob elektronisch oder auf Papier, zählen als **nicht abgegeben!**