

Einführung in die Programmierung
WS 2009/10

Übungsblatt 7: Imperative Programmierung, Parameterübergabe

Besprechung: 14./16./17./18.12.2009

Ende der Abgabefrist: Montag, 14.12.2009 10:00 Uhr.

Hinweise zur Abgabe:

Geben Sie bitte Ihre gesammelten Lösungen zu diesem Übungsblatt in einer Datei `loesung07.zip` unter <http://www.pst.ifi.lmu.de/uniworx/> ab.

Bitte beachten Sie, dass die Aufgabe 7-2 nicht in die Bonusregelung eingeht. Bereiten Sie diese aber bitte trotzdem vor, damit Sie der Übung optimal folgen können.

Aufgabe 7-1 *Sieb des Eratosthenes*

10 Punkte

Nach Eratosthenes, einem Mathematiker des alten Griechenlands, wird folgender Algorithmus zur Bestimmung aller Primzahlen bis zu einer bestimmten Zahl „Sieb des Eratosthenes“ genannt:

Angenommen, Sie wollen alle Primzahlen ≤ 1000 bestimmen.

- Erstellen Sie ein Array vom Typ `boolean[]` der Länge 1001.
- Die Elemente mit Index 0 und 1 sollen den Wert `false` haben, alle anderen zunächst den Wert `true`. (0 und 1 sind *per definitionem* keine Primzahlen.)
- Fangen Sie nun an, das Array von vorne an nach Elementen vom Wert `true` zu durchsuchen. Für jedes solche Element a an Index i_a setzen Sie den Wert aller Elemente b , deren Index i_b ein ganzzahliges Vielfaches von i_a ist, auf den Wert `false`.
- Wenn Sie die Suche beendet haben, sind die Indizes mit Wert `true` die Primzahlen ≤ 1000 .

(a) Schreiben Sie in einer Klasse `Primzahlen` eine Methode

`public static boolean[] siebDesEratosthenes(int n)`,
die für eine natürliche Zahl $n > 0$ die oben beschriebene Methode implementiert.

(b) Ergänzen Sie die Klasse um eine Methode

`public static void printPrimzahlen(int n)`,
die für eine natürliche Zahl $n > 0$ die Primzahlen $\leq n$ auf die Konsole ausgibt.

(c) Ergänzen Sie die Klasse um eine `main`-Methode, die die Methode `printPrimzahlen` mit einem `int`-Wert als Parameter aufruft.

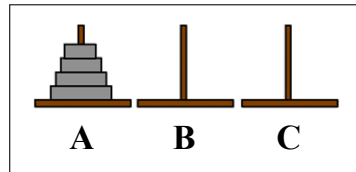
Geben Sie Ihre Lösung in einer Datei `Primzahlen.java` ab.

Aufgabe 7-2 *Türme von Hanoi*

0 Punkte

Ein bekanntes Spiel trägt den Namen „Türme von Hanoi“. Das Spiel wird mit n Scheiben verschiedener Größe gespielt, die auf Stäbe A , B , C gesteckt werden können. Zu Beginn stecken alle n Scheiben auf Stab A und zwar derart, dass jeweils eine kleinere Scheibe auf einer größeren Scheibe liegt (siehe Abbildung). Die Aufgabe besteht darin, die Scheiben von Stab A nach Stab C umzustecken; Stab B kann dabei als Zwischenspeicher verwendet werden. Es sind dabei folgende zwei Regeln zu beachten:

- In jedem Schritt darf nur genau eine Scheibe bewegt werden.
- Eine größere Scheibe darf nie auf einer kleineren Scheibe liegen.



- (a) Schreiben Sie in einer Klasse `Hanoi` eine Methode `public static stecke(int n, char A, char B, char C)`, die für eine natürliche Zahl $n > 0$ und drei Character A , B und C die oben beschriebene Methode implementiert. Die einzelnen Züge sollen dabei auf die Konsole ausgegeben werden.
- (b) Ergänzen Sie die Klasse um eine `main`-Methode, um Ihre Methode mit geeigneten Werten zu testen.

Aufgabe 7-3 *Parameterübergabe***10 Punkte**Gegeben sei die Java-Klasse `Parameteruebergabe`.

```
public class Parameteruebergabe {

    static String a = "A";

    static String b = "B";

    static String[] ab = { a, b };

    public static void swap(String s0, String s1) {
        String temp = s0;
        s0 = s1;
        s1 = temp;
    }

    public static void swap(String[] s) {
        String temp = s[0];
        s[0] = s[1];
        s[1] = temp;
    }

    public static void proverb() {
        System.out.println("Wer " + a + " sagt, muss auch " + b + " sagen!");
        System.out.println("Also muss auch " + ab[0] + " sagen, wer " + ab[1]
            + " sagt?");
    }

    public static void main(String[] args) {
        proverb();
        swap(a, b);
        proverb();
        swap(ab);
        proverb();
    }
}
```

Welche Ausgabe erzeugt das Programm? Begründen Sie Ihre Antwort ausführlich. Geben Sie Ihre Lösung in einer Datei `Parameteruebergabe.txt` ab.