

Einführung in die Programmierung

WS 2018/2019, Blatt 06

Ausgabe: 26.11.2018

Prof. Dr.-Ing. Norbert Siegmund
M.Sc. André Karge

Abgabetermin: Montag, 03.12.2018, 11:00 Uhr

Bitte lesen Sie die folgenden Informationen zum Übungsablauf **sorgfältig** durch.

Grundsätzlich – wenn nicht anders angegeben – sind die Lösungen zu den Übungen zu Einführung in die Programmierung jeden **Montag bis spätestens 11:00 Uhr** an André Karge per E-Mail zu schicken.

Schreiben Sie bitte im Betreff Ihrer E-Mail Ihre **Teamnummer** sowie die Nummer des Übungsblattes. In der E-Mail schreiben Sie bitte zusätzlich Ihren **Namen** und **Matrikelnummer**. Die Lösungen für Sie bitte als Java Dateien als Anlage hinzu. Es werden **keine** kompilierten Dateien, wie *.class oder *.jar angenommen.

Übungen müssen von **minimal zwei** und **maximal drei** Studierenden aus derselben Übungsgruppe in einem festen Team bearbeitet werden (Ausnahmen nur auf Anfrage beim Übungsleiter). Pro Team soll die Lösung nur einmal abgegeben werden. Aufgaben sollen **im Team gelöst** und nicht nur vom Team abgegeben werden. Sie müssen mindestens **50%** dieser Punkte für eine Zulassung zur Prüfung erreichen. Das **Abschreiben** identischer Lösungen wird jeweils mit 0 Punkten bewertet.

Bei Fragen oder Unklarheiten wenden Sie sich bitte **vor der Abgabe** des Übungsblattes an den Übungsleiter (per E-Mail oder persönlich). Es soll nie jemand sagen müssen: „Wir haben die Aufgabe nicht verstanden und konnten sie daher nicht bearbeiten.“

Weitere Informationen, wie aktuelle Ankündigungen, finden Sie online (<https://www.uni-weimar.de/de/medien/professuren/intelligente-softwaresysteme/lehre/>) unter Einführung in die Programmierung

Aufgabe 1 Fibonacci (7 Punkte)

Implementieren Sie die Berechnung der n -ten Fibonacci Zahl. Sie ist die Summe aus ihren beiden Vorgängern mit $fib(0) = fib(1) = 1$.

- Als rekursive Funktion.
- Als iterative Funktion.

Verwenden Sie als Rückgabewert jeweils `long`, da die Fibonacci-Zahlen sehr schnell wachsen.

Was fällt Ihnen auf, wenn Sie Ihre Implementierung testen, indem sie z. B. `fib(45)` auf beide Arten berechnen? Was sind jeweils die Laufzeiten der beiden Funktionen in \mathcal{O} -Notation?

Aufgabe 2 Foreach (6 Punkte)

Schreiben Sie ein Java Programm:

- Die Erstellung eines `double` Array, zufälliger Länge (max. 20) und Auffüllen mit zufälligen Werten, danach die Ausgabe des Arrays und die Berechnung des Durchschnitts in einer Methode `public static double calculateMean(double[] values)` sowie die Ausgabe des Durchschnitts sollen umgesetzt werden.
Verwenden Sie die Java Syntax, die dem Ausdruck `für jedes Element` entspricht, um den Durchschnitt zu berechnen.
- Schreiben Sie eine Methode `public static String reverseString(String toReverse)`, die den gegebenen String

```
String text = "DieserTextwirdrueckwaertsausgegeben . ";
```

umkehrt und als Methodenrückgabewert zurückgibt. Der umgekehrte String soll nach dem Methodenaufruf auf der Console ausgegeben werden.

Hinweise:

- `Math.random()` liefert eine Zufallszahl i mit $0.0 \leq i < 1.0$ zurück
- `int z = (int)((Math.random() * max) + 1);` liefert eine Zufallszahl zwischen 1 und max.
- Die Klasse `String` stellt die Methode `String.toCharArray()` zur Verfügung um einen `String` in ein Zeichen Array umzuwandeln.

Aufgabe 3 Palindrom (6 Punkte)

Schreiben Sie eine Funktion boolean `isPalindrom(String word)`, welche überprüft, ob das gegebene Wort ein Palindrom ist. Ein Palindrom ist ein Wort, das von vorne und rückwärts gelesen identisch ist. So liefert beispielsweise ein Aufruf von `isPalindrom("otto")` als Ergebnis `true`, `isPalindrom("ananas")` jedoch `false`. Hinweise: Die Klasse `String` bietet u.a. folgende Methoden:

- `char charAt(int index)`
Returns the character at the specified index.
 - `int length()`
Returns the length of this string.
 - `char[] toCharArray()`
Converts this string to a new character array.
-
- `int length():` liefert die Länge eines Strings
 - `char charAt(int i):` liefert das i-te Zeichen im String, beginnend bei 0
- (a) Implementieren Sie `isPalindrom` imperativ.
- (b) Implementieren Sie `isPalindrom` rekursiv.