

# Informatik 2 - Sommersemester 2018

## Übungsblatt 1

Abgabe: Montag, 30. April, 14:00 Uhr

### Aufgabe 1: Anmeldung (5 Punkte)

Melden Sie sich beim Kurssystem Daphne an. Den Link dazu finden Sie auf der Kurs-Homepage. Achten Sie darauf, dass Ihre Daten korrekt sind, insbesondere dass Sie unter der angegebenen E-Mail Adresse auch erreichbar sind. Führen sie ein checkout auf Ihr SVN-Repository

`https://daphne.informatik.uni-freiburg.de/ss2018/InformatikII/svn/[ihr_kuerzel]`

auf Daphne aus und laden Sie eine leere Datei mit Namen `python`, `cpp` oder `java` in Ihr Repository je nachdem welche Sprache Sie bevorzugen (Sie werden dann einem entsprechenden Tutor zugeteilt).

### Aufgabe 2: Quicksort (5 Punkte)

Implementieren Sie den in der Vorlesung erklärten Algorithmus *QuickSort*. Beachten Sie dazu die auf der Webseite verlinkte Design-Vorlage `QuickSort.py`.

Schreiben Sie je einen Unit Test für die `quicksort_divide` und die `quicksort_recursive` Methode. Die Unit Tests sollten grundsätzlich mindestens ein nicht-triviales Beispiel überprüfen. Wenn es kritische Grenzfälle gibt, die sich durch wenig Aufwand leicht nachprüfen lassen (z.B. Verhalten einer Methode bei leerem Eingabefeld), sollten Sie dies ebenfalls tun.

### Aufgabe 3: Zeitmessungen (5 Punkte)

Messen Sie die Laufzeit  $T(n)$  von den in der Vorlesung implementierten *SelectionSort* und *MergeSort*, sowie von Ihrer *QuickSort*-Implementierung (diese liegen im public repository, dass Sie erhalten wenn sie bei oben stehender URL ihr Kürzel mit public ersetzen). für verschiedene Eingabegrößen  $n$ . Testen Sie bei *QuickSort* zwei Varianten der Pivot-Wahl: "Element an erster Position", "Element an zufälliger Position". Stellen Sie die Laufzeit der vier Algorithmen abhängig von der Eingabegröße  $n$  graphisch dar. Wiederholen Sie das Experiment zwei Mal, wobei Sie als Eingabe einmal jeweils ein mit zufälligen Integers gefülltes Array nehmen, und einmal eine Eingabe nehmen, bei welcher die Elemente gerade verkehrt herum sortiert sind.

Vergleichen Sie die Laufzeiten der Algorithmen (alle 8 Varianten) anhand der erhaltenen Schaubilder und diskutieren Sie Ihre Ergebnisse kurz in Ihren `erfahrungen.txt` (siehe Aufgabe 4).

### Aufgabe 4: Abgabe (5 Punkte)

Committen Sie Ihren Code (inkl. Tests) und die acht Schaubilder in das SVN, in einen eigenen Unterordner `uebungsblatt-01`. Gehen Sie dabei so vor, wie in der Vorlesung vorgeführt. Stellen Sie sicher, dass auf Jenkins alles (inkl. Style Check und Unit Tests) fehlerfrei durchläuft.

Committen Sie in diesem Unterordner ausserdem eine Textdatei `erfahrungen.txt`. Beschreiben Sie dort in ein paar Sätzen Ihre Erfahrungen mit diesem übungsblatt und den Vorlesungen dazu. Insbesondere: Wie lange haben Sie ungefähr gebraucht? An welchen Stellen gab es Probleme und wieviel Zeit hat Sie das gekostet?