

# Aufgabe/Task: Nr. 08

## Thema: Runtime Behaviour

Geschätzter Zeitbedarf: 120-240 min

### Aufgabenbeschreibung:

#### Teil 1

Bearbeiten Sie diese Übungsanleitung(en):

[https://bscw.tbz.ch/bscw/bscw.cgi/d32029909/script4\\_firstStepsPerformance.pdf](https://bscw.tbz.ch/bscw/bscw.cgi/d32029909/script4_firstStepsPerformance.pdf)

[https://bscw.tbz.ch/bscw/bscw.cgi/d32029904/script4\\_ErsteSchrittePerformance.pdf](https://bscw.tbz.ch/bscw/bscw.cgi/d32029904/script4_ErsteSchrittePerformance.pdf)

Verwenden Sie diesen Code (stopwatch) um die Schnelligkeit(en) zu messen.

<https://bscw.tbz.ch/bscw/bscw.cgi/31928798>

#### Teil 2

Finden Sie in den 'normalen' Java-Board-Mitteln Datenstrukturen, mit denen Sie sammeln und sortieren lassen können (z.B. collections, arrays, u.a.) und machen Sie (statistische) Vergleiche mit dem früher implementierten BubbleSort. Verwenden Sie dabei Ihren Zahlengenerator aus der früheren Übung und nehmen Sie genügend viele Zahlen, messen Sie die Schnelligkeiten und werten Sie sie mit Excel aus.

<https://www.geeksforgeeks.org/arrays-sort-in-java-with-examples/>

#### Teil 3

Finden Sie weitere Sort-Algorithmen im Internet und implementieren Sie einen, den Sie noch nicht kennen davon und vergleichen Sie ebenfalls die Leistung (Performanz) dieses Algorithmus. (z.B. QuickSort o. a.).

**Zeigen Sie die Resultate der Lehrperson**