

Übungsserie 1 – Algorithmen und Datenstrukturen

Aufgabe 1: Messumgebung zur Laufzeitanalyse

Ziel dieser Aufgabe ist es, eine Codegerüst zu entwickeln mit dessen Hilfe Sie bei späteren Aufgaben in der Lage, Laufzeitanalysen von Anwendungen zu erstellen.

Der Code für die Laufzeitanalyse soll in der main Methode stehen um Verfälschungen der Laufzeiten durch Methodenaufrufe zu vermeiden.

Die zu erstellende Testsuite soll folgende Eigenschaften erfüllen:

- Sie soll innerhalb einer Schleife später beliebigen Code ausführen können. Die Anzahl der Schleifendurchläufe soll später variiert werden können (10.000 – 10.000.000.000).
- Die Laufzeit dieses Codes soll gemessen werden. Zur Zeitmessung kann `System.currentTimeMillis()` verwendet werden.
- Die Schleife die den Programmcode später 10.000 bis 10.000.000.000 ausführt soll insgesamt 10mal wiederholt werden. Bei jedem Durchlauf wird die Zeit gemessen und in einem Array abgespeichert.
- Anschließend soll die durchschnittliche Zeit ausgegeben werden, die jeder Durchgang benötigt hat.
- Die Ausgabe der Testsuite soll aussehen wie im Folgenden abgebildet:

```
-----  
Zeit im Durchlauf 0: 10ms  
Zeit im Durchlauf 1: 7ms  
Zeit im Durchlauf 2: 7ms  
Zeit im Durchlauf 3: 6ms  
Zeit im Durchlauf 4: 7ms  
Zeit im Durchlauf 5: 6ms  
Zeit im Durchlauf 6: 6ms  
Zeit im Durchlauf 7: 6ms  
Zeit im Durchlauf 8: 7ms  
Zeit im Durchlauf 9: 6ms  
Zeit im Durchschnitt: 6.8ms  
Die Programmschleife wurde 10000000mal ausgeführt.
```

Aufgabe 2: Messen von elementaren Operationen

Nun können Sie damit beginnen, Code in die innerste Schleife ihrer Testsuite zu schreiben. Überlegen Sie sich elementare Operationen (z.B. Wertzuweisungen) und schreiben Sie den entsprechenden Code. Variieren Sie nun den Wert für die Durchläufe der inneren Schleife (ca im Bereich von 100.000 bis 10.000.000.000) und tragen Sie die gemessenen Werte in ein Messprotokoll ein, welches z.B. wie folgt aussehen kann(!) :

n	100.000	1.000.000	10.000.000	100.000.000	1.000.000.000
Schleifendurchlauf	0,5	1,1	6,7	60,7	605,2
+ Addition von 50 Zahlen	0,6	1,2	6,8	61,5	615,9
Diff	0,1	0,1	0,1	0,8	10,7
+ 10 Schreibzugriff auf Array	0,6	1,3	6,9	61,8	620,2
Diff	0	0,1	0,1	0,3	4,3
+ 20 Divisionsrechnungen	0,9	2,6	18,8	180,1	1801,9
Diff	0,3	1,3	11,9	118,3	1181,7
+ 5 Fallunterscheidung	0,9	2,6	19,2	181,5	1811,2
Diff	0	0	0,4	1,4	9,3
+ 10 Zuweisungen	1,1	2,7	19,5	183,1	1850,1
Diff	0,2	0,1	0,3	1,6	38,9
	ms	ms	ms	ms	ms

- Was können Sie bei den Messungen feststellen?
- Trifft das ein, was Sie erwarten würden?
- Können Sie näherungsweise die Ausführungszeiten zu bestimmten Operationen ermitteln?

Aufgabe 3:

Erstellen Sie jeweils eine Methode, welche eine Laufzeit der folgenden Ordnung hat:

- 1 (konstant)
- N (linear)
- N^2 (quadratisch)
- N^3 (kubisch)

Vergleichen Sie die Laufzeiten (Messumgebung) und fertigen Sie ein Protokoll darüber an.

Aufgabe 4:

Erstellen Sie eine Methode `public static double minimum(double[] array)`, welche das übergeben Feld nach dem Minimum durchsucht und geben Sie das Minimum aus. Von welcher Ordnung ist die Laufzeit dieser Methode?