

Übungsblatt 5

Thomas Graf
EF / WF Informatik 2018-2019
Algorithmen

21. Oktober 2018

1 lineare Suche (★)

Bei der *linearen Suche* ist eine Liste $A[1...n]$ von Werten (z.B. Zahlen) gegeben sowie ein Wert v . Der lineare Such Algorithmus überprüft nun, ob der Wert v in der Liste A enthalten ist oder nicht.

Betrachte folgenden (Python-ähnlichen) Pseudocode, welcher die *lineare Suche* implementiert:

```
1 i = NIL
2 for j = 1 to A.length:
3     if A[j] = v:
4         i = j
5     return i
6 return i
```

Was ist die best-case running time für diesen Algorithmus? Was ist die worst-case running time? Beweise, unter Verwendung einer loop-Invarianten, dass der Algorithmus stimmt. Prüfe *Initialization*, *Maintenance* und *Termination*.

2 best-case running time (★★)

Wie kann man fast jeden Algorithmus so modifizieren, dass er eine sehr schnelle *best-case running time* hat?

3 Vergleich der Laufzeiten zweier Algorithmen (★★)

Wir haben zwei verschiedene Algorithmen A und B eingehend untersucht und jeweils eine Laufzeitanalyse durchgeführt. Dabei haben wir herausgefunden, dass Algorithmus A eine Laufzeit von $T_A(n) = 150n^2$ hat und Algorithmus B eine Laufzeit von $T_B(n) = 2^n$ (für gegebene Problemgrösse n). Welches ist das kleinste $n \in \mathbb{N}$, $n > 2$, sodass Algorithmus A weniger Zeit benötigt als Algorithmus B (auf demselben Computer).