

F3 – Berechenbarkeit und Komplexität

Aufgabenzettel 10: Laufzeitanalyse von Algorithmen

Besprechung in der Zeit vom 14.01. zum 18.01.2002.

Präsenzaufgabe 10:

Sei a eine Konstante. Schätzen Sie die Größenordnung von $f : \mathbb{N} \rightarrow \mathbb{N}$ mit $f(n) := n \cdot \sum_{i=1}^n (i + a^i)$ ab. Unterscheiden Sie die Fälle $a = 1$ und $a \neq 1$.

Übungsaufgabe 10.1:

Seien $a > 0$ und b Konstanten und $f_n := f_{n-1} + a \cdot f_{n-1} - b$.

1. Leiten Sie eine geschlossene Formel für f_n her. (4 Pkt.)
2. Nennen Sie eine praktische Anwendung für diese Rekurrenz! (2 Pkt.)

von
6

Übungsaufgabe 10.2:

Seien die Elemente des Feldes s aufsteigend sortiert und geben **low** und **high** den unteren bzw. oberen Index des zu durchsuchenden Bereichs an. Der zu suchende Wert sei c . Betrachten Sie den folgenden Algorithmus:

```
procedure contains(S, low, high, c): boolean
  if low > high
    then return false;
  else m := (low + high) div 2;
  if S[m] = c
    then return true;
  else if S[m] < c
    then return contains(S, m+1, high, c);
  else return contains(S, low, m-1, c);
```

von
6

Analysieren Sie die Zeitkomplexität des Algorithmus, indem Sie $T(1)$ angeben und eine Rekurrenzgleichung für $T(n)$ angeben. Lösen sie die Rekurrenzgleichung auf, um zu einer geschlossenen Form und somit zu einer Komplexitätseinstufung in O -Notation zu kommen.

Bisher erreichbare Punktzahl:

121

Hinweis: Sie dürfen bei der Lösung der Aufgaben folgendes Faktum verwenden:

$$\sum_{i=0}^n x^i = \frac{x^{n+1} - 1}{x - 1} \quad (\text{geometrische Reihe})$$