

Informatik 1

13. Übung

Aufgabe 1 (Rekursionsgleichung)

Führen Sie die Methode aus der Vorlesung durch, um eine Lösung der Rekursionsgleichung

$$T(n) = T(n - 1) + 2n - 1, \quad T(1) = 1$$

in geschlossener Form zu entwickeln.

Berechnen Sie zur Probe $T(5)$ mit Hilfe Ihrer gefundenen Formel und mit Hilfe der Rekursionsformel.

Aufgabe 2 (Ackermann-Funktion)

Gegeben sei die in zwei Variablen rekursiv definierte Funktion

$$\begin{aligned} a(0, m) &= m + 1 \\ a(n + 1, 0) &= a(n, 1) \\ a(n + 1, m + 1) &= a(n, a(n + 1, m)) \\ m, n &\in \mathbf{N} \end{aligned}$$

a) Rechnen Sie die folgenden Ausdrücke aus:

- i) $a(0, 9)$
- ii) $a(1, 1)$
- iii) $a(1, 2)$
- iv) $a(2, 1)$
- v) $a(2, 2)$

b) Zeigen Sie, dass folgende Formeln gelten:

- i) $a(1, m) = 2 + m$
- ii) $a(2, m) = 2m + 3$
- iii) $a(3, m) = 8 \cdot 2^m - 3$

Aufgabe 3 (Rekursive Maximum-Funktion)

Schreiben Sie eine rekursive Maximum-Funktion für ein Array, die die in der Vorlesung vorgestellte Idee (Halbierung des Arrays) implementiert.

Die Funktionsdeklaration hierfür ist

```
integer max(integer a[ ], integer left, integer right)
```

und es bedeutet, dass das Maximum für

```
a[left], a[left+1], a[left+2], ..., a[right]
```

ausgerechnet werden soll.

Der Benutzer ruft diese Funktion für ein n -elementiges Array immer mit

```
max(a[ ], 0, n-1)
```

auf.

Abgabe: Freitag, 5. Februar 2016