ETH Zürich
Institut für Theoretische Informatik
Prof. Dr. Angelika Steger, Dr. Johannes Lengler
Florian Meier

HS 2019
Abgabe 24.09.2018

Algorithmen und Komplexität
Übungsblatt 1

Zeit, Ort und Einteilung der Übungsgruppen sowie weitere Informationen stehen auf der Homepage der Vorlesung (http://www.cadmo.ethz.ch/education/lectures/HS19/ac). Bitte lesen Sie auch alle Informationen auf der Homepage bezüglich Abgabe und Korrektur der Übungsblätter durch.

Bei Fragen zur Vorlesung oder zu den Übungen wenden Sie sich bitte direkt an die Übungsleitung (meierflo@inf.ethz.ch).

⋆ ⋆ ⋆

In dieser Serie werden alle Aufgaben von den Assistenten korrigiert.

⋆ ⋆ ⋆

Aufgabe 1

Entscheiden Sie, ob folgende Aussagen zutreffen:

(a) $\ln n = \Theta(\log_2 n)$, wobei ln den natürlichen Logarithmus (zur Basis e) bezeichnet,

(b) $n = \mathcal{O}(\log_2 n)$,

(c) $n! = \mathcal{O}(n^n)$,

(d) $n! = \Theta(n^n)$,

(e) $2^{2n} = \mathcal{O}(2^n)$,

(f) $1/n = \mathcal{O}(1)$.

Aufgabe 2

Gegeben sei eine RAM-Maschine. Wir nehmen an, dass in der nullten Speicherzelle $M_0 = n$ gegeben ist und in den Speicherzellen $M_1, \ldots, M_n$ natürliche Zahlen stehen. Vervollständigen Sie die folgenden Zeilen zu einem Programm, das die Zahlen in $M_1, \ldots, M_n$ verdoppelt.

Eingabe: n natürliche Zahlen in $M_1, \ldots, M_n$ sowie $M_0 = n$.
Ausgabe: Die Zahlen in $M_1, \ldots, M_n$ verdoppelt.
1: $M_{-2} \leftarrow 1$
2: $M_{-1} \leftarrow M_{M_0}$
3: $M_{-1} \leftarrow M_{-1} + M_{-1}$
...

Aufgabe 3

In der Vorlesung haben Sie zwei Varianten für die Analyse der Laufzeit eines Algorithmus kennengelernt, das Einheitskostenmass und das logarithmische Kostenmass. Im Einheitskostenmass wird angenommen, dass jede arithmetische Operation Laufzeit 1 benötigt. Im logarithmischen Kostenmass wird angenommen, dass jede arithmetische Operation die Summe der Längen der Binärdarstellungen der involvierten Zahlen an Zeit benötigt. Das heisst, dass zum Beispiel die Addition oder Multiplikation zweier Zahlen n und m unter Annahme des Einheitskostenmasses 1 Zeit benötigt und unter Annahme des logarithmischen Kostenmasses $\lceil \log_2(n) \rceil + \lceil \log_2(m) \rceil$

Zeit benötigt. Das Ziel dieser Aufgabe ist es drei Algorithmen, welche auf unterschiedliche Weise die n-te Fibonacci-Zahl berechnen, sowohl mit dem Einheitskostenmass als auch mit dem logarithmischen Kostenmass zu analysieren.

Die Fibonacci-Zahlen F_n sind folgendermassen definiert:

$$F_0 := 0, \quad F_1 := 1 \quad \text{und} \quad F_n := F_{n-1} + F_{n-2} \text{ für } n \geq 2.$$

Sei $n = 2^k$ eine Zweierpotenz. Die folgenden Algorithmen FIBA, FIBB und FIBC berechnen die n-te Fibonacci-Zahl (Machen Sie sich klar dass auch FIBC die n-te Fibonacci-Zahl berechnet). Nehmen Sie für folgende Aufgaben an, dass die Laufzeit der Algorithmen proportional zur Zeit, welche für die Berechnung der arithmetischen Operationen werwendet wird, ist.

(a) Vergleichen Sie die Laufzeiten der drei Algorithmen unter Annahme des Einheitskostenmasses.

(b) Vergleichen Sie die Laufzeiten der drei Algorithmen unter Annahme des logarithmischen Kostenmasses.

(c) Welches Kostenmass ist für diese Algorithmen die vernünftigere Wahl?

(d) Beschreiben Sie in Worten, wie Algorithmus FIBC modifiziert werden kann, sodass er auch die n-te Fibonacci-Zahl berechnet, wenn n keine Zweierpotenz ist. Die asymptotische Laufzeit soll dabei unverändert bleiben.

Algorithm 1 FIBA(n)

```
if n = 0 then
    return 0
else if n = 1 then
    return 1
else
    return FIBA(n − 1) + FIBA(n − 2)
end if
```

Algorithm 2 FIBB(n)

```
if n ≤ 1 then
    return n
end if
A1 := 0
A2 := 1
F := 0
for i = 2 to n do
    F := A1 + A2
    A1 := A2
    A2 := F
end for
return F.
```

ABGABE DER HAUSAUFGABEN IN DER VORLESUNG AM 24.09.2018.

Algorithm 3 FIBC(n)

if $n \leq 1$ **then**
  return n
end if
$k := \log_2(n)$
$A := \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$
for $i = 1$ **to** k **do**
  $A := A^2$
end for
return A_{12}.