Institut für Informatik I
Prof. Dr. Heiko Röglin
Tobias Brunsch

Algorithmen und Berechnungskomplexität I
Wintersemester 2010/11
Abgabe: 20.10.2010 in der Vorlesung

Übungsblatt 1

Aufgabe 1.1

1+3+2 Punkte

Beweisen Sie folgende Aussagen.

(a) $\forall k > 0\colon n^k = o(2^n)$

(b) Es seien $p_1(n) = a_1 \cdot n^{d_1}$ und $p_2(n) = a_2 \cdot n^{d_2}$ Polynome vom Grad d_1 bzw. d_2, wobei die Koeffizienten a_1 und a_2 positiv sind. Dann gilt:

(I) $p_1 = \Theta(p_2) \iff d_1 = d_2$

(II) $p_1 = o(p_2) \iff d_1 < d_2$

(III) $p_1 = \omega(p_2) \iff d_1 > d_2$

(c) $\forall k > 0\ \forall \varepsilon > 0\colon (\log_2(n))^k = o(n^\varepsilon)$

Hinweis: Für zwei Funktionen $f\colon \mathbb{N} \longrightarrow \mathbb{R}^+$ und $g\colon \mathbb{N} \longrightarrow \mathbb{R}^+$ gilt $f = o(g) \iff \lim\limits_{n\to\infty} \frac{f(n)}{g(n)} = 0$.

Aufgabe 1.2

5+1 Punkte

Füllen Sie die folgende Tabelle aus. Verwenden sie eines der Symbole aus $\{O, o, \omega, \Omega, \Theta, -\}$, um eine Funktion f einer Zeile mit einer Funktion g einer Spalte in Beziehung zu setzen. Versuchen Sie, so genau wie möglich zu sein. Beispiel: Für $f(n) = n^3$ und $g(n) = n^4$ gilt $f = o(g)$. Der entsprechende Eintrag sollte also „o“ und nicht nur „O“ sein. Stehen die Funktionen in keiner Beziehung, tragen Sie „$-$“ ein.

In der Tabelle stehe $s(n)$ abkürzend für die Funktion $s(n) = \begin{cases} 1 & : \ n \text{ ungerade}, \\ n & : \ n \text{ gerade}. \end{cases}$

	$\log_2(n)$	$s(n)$	$\sqrt{n}$	5	2^n	$1/n$	n	e^n	n^2
$\log_2(n)$									
$s(n)$									
$\sqrt{n}$									
5									
2^n									
$1/n$									
n									
e^n									
n^2									

Hinweis: Es genügt, einen der beiden Teile unterhalb oder oberhalb der Diagonale auszufüllen und zu beschreiben, wie sich die restlichen Einträge ergeben.

Aufgabe 1.3

3+3 Punkte

(a) Schreiben Sie ein Registermaschinenprogramm, das aus der Eingabe $n \in \mathbb{N}_0$ den Wert

$$g(n) = \sum_{i=1}^{n} i^2$$

berechnet. Die Eingabe n befinde sich in Register 1. Das Ergebnis $g(n)$ soll in Register 2 gespeichert werden. Beschreiben Sie die Aufgaben der verwendeten Register und kommentieren Sie die wesentlichen Schritte Ihres Programms.

(b) Gegeben sei das folgende Registermaschinenprogramm, welches zu jeder Eingabe $n \in \mathbb{N}_0$ einen Funktionswert $f(n)$ berechnet. Das Ergebnis befindet sich am Ende der Berechnung in Register 4.

Untersuchen Sie, welche Funktion f berechnet wird. Gehen Sie dabei insbesondere auf die Bedeutung der verwendeten Register ein. Versuchen Sie zudem, die Anzahl der Rechenschritte abzuschätzen (O-Notation).

```
// Eingabe n steht in Register 1
 1 CLOAD 1
 2 STORE 2
 3 STORE 3
 4 STORE 4
 5 LOAD 1
 6 IF c(0) < 2 GO TO 18
 7 LOAD 2
 8 ADD 3
 9 STORE 4
10 LOAD 3
11 STORE 2
12 LOAD 4
13 STORE 3
14 LOAD 1
15 CSUB 1
16 STORE 1
17 GO TO 6
18 END
// Ergebnis f(n) steht in Register 4
```

Aufgabe 1.4

2+2+2 Punkte

Sei A ein Feld mit den Einträgen $1, \ldots, n$ in beliebiger Reihenfolge. Betrachten Sie folgenden Algorithmus.

Eingabe: A
for $i = 1, \ldots, n-1$ **do**
  Bestimme das Minimum der Einträge $A[i], \ldots, A[n]$ und den zugehörigen Index j.
  Falls $j \neq i$, dann vertausche $A[i]$ und $A[j]$.
end for
Ausgabe: A

(a) Welches Problem löst der Algorithmus? Beweisen Sie Ihre Aussage mit Hilfe einer Invariante.

(b) Wie viele Vergleiche werden in Durchlauf i durchgeführt? Wie viele Vergleiche benötigt der Algorithmus insgesamt? Verwenden Sie O-Notation.

(c) Wie viele Vertauschungen führt der Algorithmus mindestens/höchstens durch? Geben Sie jeweils eine Eingabe an, bei der entsprechend oft vertauscht wird.