



CHRISTIAN-ALBRECHTS-UNIVERSITÄT ZU KIEL

Institut für Informatik, Arbeitsgruppe Theorie der Parallelität
Prof. Dr. K. Jansen, K.-M. Klein, M.Maack

15. Mai 2015

Hausaufgaben zur Vorlesung »Algorithmen und Datenstrukturen«

Blatt 5

Hausaufgabe 5.1 (Rekurrenzgleichung (2 Punkte))

Gegeben seien die folgenden Rekurrenzgleichungen für $a, b \in \mathbb{N}$:

- (a) $T(0) = a, T(n) = T(n-1) + b$ für alle $n \in \mathbb{N} \setminus \{0\}$
- (b) $T(1) = 1, T(n) = T(\frac{n}{2}) + n$ für alle $n \in \mathbb{N}$ derart, dass es ein $k \in \mathbb{N} \setminus \{0\}$ gibt mit $n = 2^k$.

Zeigen Sie, dass in beiden Fällen $T(n) \in \mathcal{O}(n)$ gilt.

Hausaufgabe 5.2 (Fibonacci Zahlen (2 Punkte))

Beweisen Sie, für alle $n \in \mathbb{N} \setminus \{0, 1\}$

$$F_n \geq \left(\frac{1 + \sqrt{5}}{2}\right)^{n-2},$$

wobei F_n die n -te Fibonacci Zahl ist.

Hausaufgabe 5.3 (Dynamische Programmierung (3 Punkte))

Überlegen Sie sich die Rekursionsvorschrift für ein dynamisches Programm für das folgendes Problem:

Sie sind ein Erasmus Student und müssen Kurse aus einem Angebot von K_1, \dots, K_n Kursen belegen. Dabei hat jeder Kurs K_i eine gewisse Zahl $E_i \in \mathbb{N}$ von ECTS Punkten und einen Aufwand $A_i \in \mathbb{N}$ an Arbeitsstunden. Gesucht ist eine Auswahl von Kursen so dass mindestens C ECTS Punkte (wobei $C \in \mathbb{N}$) erreicht werden und gleichzeitig ihr gesamter Arbeitsaufwand für die ausgewählten Kurse minimal ist.

Überlegen Sie sich wie man aus der Rekursionsvorschrift einen Algorithmus zur Berechnung einer optimalen Lösung angeben kann.

Hausaufgabe 5.4 (Programmierung (3 Punkte))

Gegeben sei im Rucksackproblem eine Menge von n Items. Jedes Item i hat dabei eine Größe gegeben durch einen Arrayeintrag $S[i]$ des Typs `int` und einen Profit gegeben durch einen Arrayeintrag $P[i]$ des Typs `int`. Weiter sei die Kapazität des Rucksacks durch eine Variable K des Typs `int` gegeben. Implementieren Sie das in der Vorlesung vorgestellt dynamische Programm, um den größtmöglichen Profit von Items zu berechnen, die von ihrer Größe her in den Rucksack passen. Erweitern Sie das dynamische Programm um eine entsprechende Teilmenge von Items zu bestimmen, welche maximalen Profit ergibt.

Verwenden Sie die von uns bereitgestellten Dateien und testen Sie Ihr Programm mit dem bereitgestellten Testprogramm. Verwenden Sie zur Ausführung des Testprogramms den Befehl `"python test.py Rucksack <Pfad>"`. Abgaben, die sich nicht mit dem Testprogramm testen lassen, werden mit 0 Punkten bewertet.

Abgabe der theoretischen Aufgaben Donnerstag, den 21. Mai, bis spätestens 11 Uhr im Schrein. Die Abgabe der Programmieraufgaben erfolgt per EMail an Ihren Übungsleiter.