Technische Universität Braunschweig
Institut für Betriebssysteme und Rechnerverbund
Abteilung Algorithmik

Sommersemester 2013

Prof. Dr. Sándor P. Fekete
Stephan Friedrichs

Klausur
Algorithmen und Datenstrukturen II
29. Juli 2013

Name: ....

Vorname: ....

Matr.-Nr.: ....

Studiengang: ....

Bachelor ☐ Master ☐ Diplom ☐ Andere ☐

Mit der Veröffentlichung des Klausurergebnisses nur mit der Matrikelnummer über die Mailingliste und auf der Homepage bin ich einverstanden.

....
Unterschrift

Hinweise:

- Bitte das Deckblatt vollständig ausfüllen.
- Die Klausur besteht aus 14 Blättern, bitte auf Vollständigkeit überprüfen.
- Erlaubte Hilfsmittel: Keine.
- Eigenes Papier ist nicht erlaubt.
- Die Rückseiten der Blätter dürfen beschrieben werden.
- Die Heftung der Blätter darf nicht entfernt werden.
- Die Klausur ist mit 50% der Punkte bestanden.
- Antworten die *nicht* gewertet werden sollen bitte deutlich durchstreichen.
- Mit *Bleistift* oder in *Rot* geschriebene Klausurteile können nicht gewertet werden.
- Werden mehrere Antworten gegeben, werten wir die mit der geringsten Punktzahl.
- Die Bearbeitungszeit für die Klausur ist 120 Minuten.

Aufgabe	**1**	**2**	**3**	**4**	**Σ**
Punkte	**30**	**20**	**25**	**25**	**100**
Erzielte Punkte					

Aufgabe 1: Knapsack **(5+5+10+10 Punkte)**

Gegeben ist folgende BINARY-KNAPSACK-Instanz mit 6 Gegenständen:

i	1	2	3	4	5	6
Wert p_i	9	1	5	3	6	2
Gewicht w_i	6	4	6	4	5	1

Dabei darf der Rucksack mit einem Gesamtgewicht von maximal $Z = 11$ bepackt werden.

a) Ermittle mithilfe des Greedy-Algorithmus aus der Vorlesung eine zulässige Lösung für die gegebene Instanz. Gib in jeder Iteration den aktuellen Gegenstand an, sowie die Menge der bereits gepackten Gegenstände mitsamt ihrem Gesamtgewicht und -wert.

b) Interpretiere die oben gegebene Instanz als Instanz des FRACTIONAL-KNAPSACK-Problems und berechne mithilfe des Greedy-Algorithmus aus der Vorlesung eine zulässige Lösung. Gib dabei in jeder Iteration den aktuellen Gegenstand, zu welchen Teilen er gepackt wird, sowie den Gesamtzustand (Was ist ist gepackt und wie hoch ist der aktuelle Gesamtwert und -gewicht?) an. Ist diese Lösung optimal? Begründe deine Aussage.

c) Wende den Dynamic-Programming-Algorithmus aus der Vorlesung auf die gegebene Instanz an. Dabei soll nicht nur der maximal erreichbare Wert, sondern auch die Menge der dafür zu packenden Gegenstände algorithmisch ermittelt werden.

   Gib die dafür nötige Tabelle komplett mit allen Einträgen an und bearbeite die Gegenstände in der Reihenfolge ihrer Indizes. Immer wenn es für den erreichten Gesamtwert keinen Unterschied macht, ob ein Gegenstand gepackt wird oder nicht, bevorzuge ,,packen".

d) Betrachte jetzt Branch and Bound.

(i) Welche untere Schranke lässt sich aus a) ableiten? Für welche Teilbäume gilt sie?

(ii) Welche obere Schranke lässt sich aus b) ableiten? Für welche Teilbäume gilt sie?

(iii) Welche obere Schranke lässt sich für den Teilbaum mit $x_1 = 1$ und $x_2 = 1$ angeben? Was lässt sich daraus im Hinblick auf die aus den vorigen Teilaufgaben bekannten Schranken folgern?

(iv) Welche untere Schranke lässt sich für den Teilbaum mit $x_1 = 1$, $x_2 = 0$, $x_3 = 0$, $x_4 = 0$, $x_5 = 1$ angeben? Was lässt sich daraus im Hinblick auf die aus den vorigen Teilaufgaben bekannten Schranken folgern?

Aufgabe 2: Komplexität **(10+4+6 Punkte)**

Gegeben sei die folgende Instanz von 3SAT:

$$S = (x_1 \vee x_2 \vee x_4) \wedge (\bar{x}_1 \vee x_3 \vee \bar{x}_4) \wedge (x_2 \vee \bar{x}_3 \vee \bar{x}_4) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_4)$$

a) Reduziere S auf eine Instanz I des BINARY-KNAPSACK-Problems und einen Wert P, so dass I genau dann eine Lösung vom Wert P oder mehr hat, wenn S erfüllbar ist.

b) Welche Konsequenzen hat es für konkrete Lösungsansätze, dass BINARY KNAPSACK NP-vollständig ist?

c) Gib drei prinzipiell verschiedene Vorgehensweisen für ein NP-vollständiges Problem an, benenne jeweils die Vor- und Nachteile jeder Alternative und führe jeweils ein Beispiel für solch einen Algorithmus an. (Hinweis: Es reicht, die Algorithmen zu benennen, sie müssen nicht komplett aufgeschrieben werden.)

Aufgabe 3: Approximationsalgorithmen **(5+5+5+5+5 Punkte)**

Wir betrachten das Bin-Packing-Problem. Gegeben sind Items $I = \{1, \dots, n\}$, wobei Item $i \in I$ Größe $0 < s_i \leq 1$ hat, und Bins $B = \{B_1, B_2, \dots\}$, jeweils mit Kapazität 1. Gesucht ist eine Zuordnung der Items in möglichst wenige Bins; dabei darf kein Bin Items einer Gesamtgröße von mehr als 1 enthalten.

Algorithmus 1, NextFit, packt der Reihe nach Items in den aktuellen Bin, bis das nächste Item nicht mehr passt. Danach wird ein neuer Bin geöffnet.

```
function NextFit(s_1, ..., s_n)
    b ← 1                                        ▷ current bin
    B_1 ← ∅

    for all i ∈ {1, ..., n} do
        if s_i + \sum_{j ∈ B_b} s_j > 1 then
            b ← b + 1
            B_b ← ∅                              ▷ open new bin
        end if
        B_b ← B_b ∪ {i}
    end for

    return b
end function
```

Algorithmus 1: NextFit Approximation für Bin-Packing

a) Wende Algorithmus 1 auf folgende Eingabe an: $s_1 = 0.3, s_2 = 0.4, s_3 = 0.7, s_4 = 0.4, s_5 = 0.1, s_6 = 0.6, s_7 = 0.2, s_8 = 1.0$. Gib dabei in jeder Iteration an, welches Item welchem Bin zugeordnet wird, sowie seinen neuen Füllstand. Liste nach Anwendung des Algorithmus' auf, welcher Bin welchen Füllstand hat und welche Items enthält.

b) Konstruiere für beliebige $n \in \mathbb{N}_{\geq 1}$ eine Sequenz von $2n + 1$ Items, so dass die optimale Verteilung k^* Bins, Algorithmus 1 aber $2k^* - 1$ Bins benötigt. (Hinweis: Betrachte abwechselnd kleine und große Items.)

c) Zeige: Algorithmus 1 hat polynomielle Laufzeit.

d) Zeige: Algorithmus 1 liefert eine 2-Approximation. Verwende dabei, dass $\sum_{i=1}^{n} s_i$ eine untere Schranke für die Optimallösung ist. (Hinweis: Wie gut füllt Algorithmus 1 zwei aufeinanderfolgende Bins?)

e) Skizziere durch eine geeignete Reduktion, dass aus der Existenz eines α-Approximationsalgorithmus $\mathcal{A}$ für Bin-Packing mit $1 \leq \alpha < \frac{3}{2}$ die Konsequenz $P = NP$ folgt.

Aufgabe 4: Standortprobleme **(3+6+10+6 Punkte)**

Der Median m einer Menge $\{x_1, \ldots, x_n\}$ erfüllt definitionsgemäß

$$|\{x_i \mid x_i < m\}| \leq \frac{n}{2} \quad \text{und} \quad |\{x_i \mid x_i > m\}| \leq \frac{n}{2}.$$

a) Betrachte die Menge $M = \{42, 7, 3, 8, 14, 33\}$. Gib die Menge aller Mediane von M an!

b) Was passiert mit der Laufzeit des Algorithmus ,,Median der Mediane", wenn man jeweils Dreier- statt Fünfergruppen verwendet?

   Was passiert mit der Laufzeit, wenn man Siebener- statt Fünfergruppen verwendet?

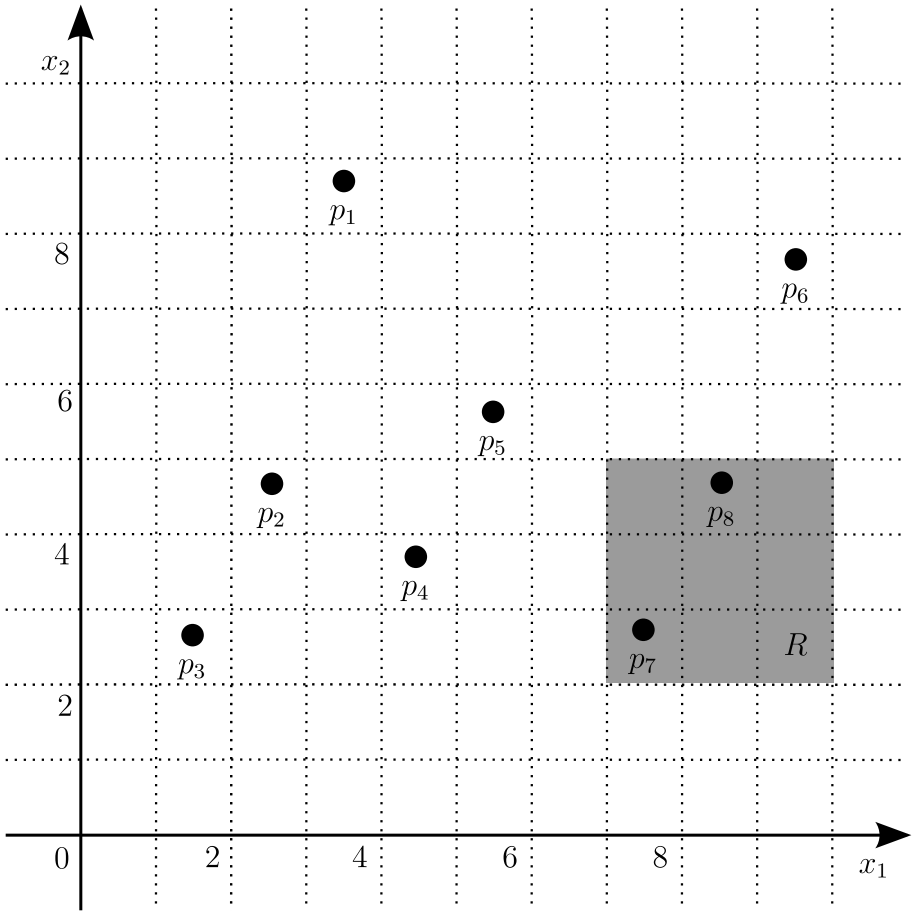

Abbildung 1: Die Punktmenge P und das Rechteck R

c) Zeichne in Abbildung 1 eine Menge von achsenparallelen Unterteilungen ein, die einen zweidimensionalen KD-Baum darstellen. Unterteile dabei zunächst nach x_1-Koordinate. Zeichne den entsprechenden KD-Baum.

d) Betrachte nun das Rechteck $R = [7, 10] \times [2, 5]$. Markiere sämtliche Knoten im KD-Baum nach folgenden Kriterien für den jeweiligen Teilbaum:

(E) R enthält das gesamte Gebiet für einen Teilbaum.

(D) R und das Gebiet für einen Teilbaum sind disjunkt.

(S) R und das Gebiet für einen Teilbaum schneiden sich, enthalten sich aber nicht gegenseitig.

Viel Erfolg ☺