

Vorlesung „Algorithmen und Datenstrukturen“

Sommersemester 2008

2. Übungsblatt

1. Greedy-Strategie (15 Punkte)

Der Manager einer Studentenvereinigung hat folgendes Problem. Gegeben sei eine Gruppe von n Studenten, von denen jeder innerhalb einer Woche für genau eine Schicht eingeteilt wird. Mit jeder Schicht sind unterschiedliche Tätigkeiten verbunden und jede Schicht ist durch ein Zeitintervall bestimmt, das unterschiedlich lang sein und an einem unterschiedlichen Zeitpunkt beginnen kann im Vergleich zu den anderen Schichten. Zu einem Zeitpunkt kann es mehrere Schichten geben. Eine Teilmenge der n Studenten soll ein *aufsichtsführendes Komitee* bilden, mit dem sich der Manager einmal pro Woche trifft. Das Komitee wird als *komplett* angesehen, falls für jeden Studenten, der nicht zum Komitee gehört, ein Student im Komitee ist, so dass sich die Schichten der beiden Studenten zumindest teilweise überschneiden. Auf diese Art und Weise kann die Leistung eines jeden Studenten durch eine Person im Komitee beobachtet werden.

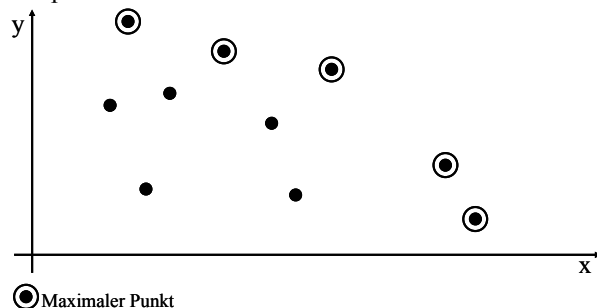
- Geben Sie einen Greedy-Algorithmus im Pseudocode an, der für n Schichten ein komplettes Komitee aus möglichst wenigen Studenten bestimmt. (8 Punkte)
- Bestimmen Sie die Laufzeit des Algorithmus. (5 Punkte)
- Beweisen Sie die Optimalität des Algorithmus. (2 Punkte)

Bemerkung: Es reicht, entsprechende Intervalle zu finden, aus denen dann jeweils ein Student in das aufsichtsführende Komitee aufgenommen wird.

2. Teile und Herrsche (12 Punkte)

Wir betrachten Punkte in der Ebene. Für zwei Punkte $p_1 = (x_1, y_1)$ und $p_2 = (x_2, y_2)$ sagen wir, p_1 dominiert p_2 , wenn $x_1 \geq x_2$ und $y_1 \geq y_2$. Ein Punkt heißt maximal, wenn er von keinem anderen Punkt dominiert wird (siehe auch unten angegebenes Beispiel). Geben Sie einen Teile-und-Herrsche Algorithmus im Pseudocode an, der die maximalen Punkte einer Punktmenge der Größe n berechnet. Analysieren Sie die Laufzeit des Algorithmus.

Beispiel:



3. Dynamisches Programmieren (12 Punkte)

Wir betrachten noch einmal das Problem, das bereits in den Präsenzaufgaben zum zweiten Übungsblatt gestellt worden ist:

Die Besatzung eines Raumschiffs findet auf einem entfernten Planeten folgende 3 Gegenstände:

Gegenstand	Gewicht (in kg)	Wert
1	10	60
2	20	100
3	30	120

Man möchte Gegenstände so mitnehmen, dass der Profit möglichst hoch ist. Das Raumschiff kann aber nur noch 50 kg transportieren, so dass man nicht alle 3 Gegenstände mitnehmen kann, sondern höchstens 2 davon.

- Entwickeln Sie einen Algorithmus in Pseudocode, der dynamisches Programmieren verwendet. Der Algorithmus soll dabei eine beliebige Anzahl von Gegenständen verarbeiten können.
- Geben Sie den Inhalt der verwendeten Datenstruktur nach jedem Schritt von Ihrem Algorithmus an.

Bemerkungen:

- Jede Seite soll oben rechts den Namen der Abgebenden und die Übungsgruppennummer (wichtig!) enthalten.
- Lösungen für die Übungsaufgaben sind (in der Regel) zu zweit abzugeben.
- Kommentieren Sie Ihre Lösungen! Besteht eine Lösung aus mehreren Zetteln, so sind diese zusammen zu heften. Bitte keine Hüllen, Mappen, o.ä..
- Bitte schicken Sie *Programmieraufgaben zusätzlich zur Abgabe auf Papier in elektronischer Form per Email* an Ihren jeweiligen Tutor.
- Kommentieren Sie ihren Quelltext bei Programmieraufgaben. Dabei sollen keine Trivialitäten kommentiert werden, also bitte keine Kommentare wie

~~x=5; // Wir weisen nun der Variablen x den Wert 5 zu~~

sondern sinnvolle Kommentare, die Ideen des Quelltextabschnittes beschreiben oder auf Unteraufgaben (z. B. a), b), ...) hinweisen.

- **Hinreichende Bedingung für die Zulassung zur Klausur:** 50% der erreichbaren Punkte bei jedem Übungszettel (bis auf zwei) und einmaliges Vorrechnen in der Übung
- **Zertifikatskriterium:** Das Bestehen der Klausur am Ende des Semesters

Abgabetermin: Donnerstag, 24.4.2008, nach der Vorlesung