

Institut für Theoretische Informatik
Peter Widmayer
Thomas Tschager
Antonis Thomas

6. April 2016

Datenstrukturen & Algorithmen

Blatt 6

FS 16

Hinweis zur Beschreibung eines dynamischen Programms: Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte (interessant auch für die Klausur!):

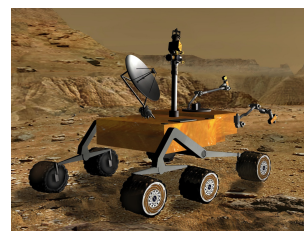
- 1) *Definition der DP-Tabelle:* Welche Dimensionen hat die Tabelle? Was ist die Bedeutung jedes Eintrags?
- 2) *Berechnung eines Eintrags:* Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?
- 3) *Berechnungsreihenfolge:* In welcher Reihenfolge kann man die Einträge berechnen, so dass die jeweils benötigten anderen Einträge bereits vorher berechnet wurden?
- 4) *Auslesen der Lösung:* Wie lässt sich die Lösung am Ende aus der Tabelle auslesen?

Die Laufzeit eines dynamischen Programms berechnet sich üblicherweise einfach aus der Grösse der Tabelle multipliziert mit dem Aufwand, jeden Eintrag zu berechnen. Manchmal überwiegt jedoch auch der Aufwand um die Lösung auszulesen.

Aufgabe 6.1 *Marsmission.*

Der Rover *Curiosity* ist auf dem Mars gelandet und befindet sich auf einer Startposition S . Das Ziel ist eine Position Z , und auf dem Weg dorthin sollen möglichst wertvolle Gesteinsproben gesammelt werden. Um nicht zu viel Energie zu verbrauchen, darf der Rover nur Schritte nach Osten (rechts) und nach Süden (unten) ausführen. Der Wert jeder Gesteinsprobe ist in einer $(m \times n)$ -Matrix gespeichert, z.B.

S	9	2	5	11	8
17	21	32	5	15	3
2	2	3	8	1	5
8	2	8	11	15	9
0	5	3	10	4	Z



In der obigen Matrix ist ein Süd-Ost-Weg von S nach Z eingezeichnet, auf dem der Wert der gesammelten Gesteinsproben maximal ist. Dieser kann durch die Angabe der benutzten Positionen beschrieben werden: $(1, 1) \rightarrow (2, 1) \rightarrow (2, 2) \rightarrow (2, 3) \rightarrow (2, 4) \rightarrow \dots \rightarrow (5, 6)$.

Geben Sie einen Algorithmus der *dynamischen Programmierung* an, der als Eingabe eine $(m \times n)$ -Matrix A mit $A[1, 1] = A[m, n] = 0$ erhält und einen Süd-Ost-Weg von $S = (1, 1)$ nach $Z = (m, n)$ berechnet, auf dem die gesammelten Gesteinsproben maximalen Wert besitzen. Gesucht wird also der Weg selbst und nicht nur der maximal zu erreichende Wert. Geben Sie auch die Laufzeit Ihres Verfahrens in Abhängigkeit von m und n an.

Bitte wenden.

Aufgabe 6.2 *Selbstanordnende Listen.*

Geben Sie an, wie viele Schlüsselvergleiche benötigt werden, wenn in der folgenden selbstanordnenden Liste mit der Move-to-Front-Regel auf die Elemente 'R', 'A', 'I', 'S', 'T', 'A', 'A', 'G' in dieser Reihenfolge zugegriffen wird.

A → L → G → O → R → I → T → H → M → U → S

Aufgabe 6.3 *Tauschgeschäft (Programmieraufgabe).*

Sie haben n verschiedene Arten von Gegenständen zur Verfügung. Jede Menge von drei Gegenständen verschiedener Arten kann in 1 Franken umgetauscht werden. Wie viele Franken können Sie maximal erhalten, wenn es a_i Gegenstände der Art i gibt?

Eingabe Die erste Zeile der Eingabe enthält lediglich die Anzahl der Testfälle t . Danach folgen t Testfälle, die wie folgt repräsentiert werden: Die erste Zahl enthält die Anzahl verschiedener Arten von Gegenständen $n \leq 5000$. Danach folgen n Ganzzahlen $a_i \leq 2000$ mit $1 \leq i \leq n$, die die Anzahl der Gegenstände der Art i angeben.

Ausgabe Für jeden Testfall soll ausgegeben werden, wie viele Franken höchstens getauscht werden können.

Beispiel

Eingabe:

```
3
2 5 10
3 2 3 4
4 1 2 3 4
```

Ausgabe:

```
0
2
3
```

Bemerkungen Für diese Aufgabe gibt es drei Kategorien von Testfällen, für die insgesamt 100 Punkte vergeben werden:

- **klein** (40 Punkte): Für dieses Testset können Sie $n \leq 20$ annehmen.
- **mittel** (40 Punkte): Für dieses Testset können Sie $n \leq 1000$ annehmen.
- **gross** (20 Punkte): Für dieses Testset gibt es keine zusätzlichen Annahmen.

Abgabe: Am Mittwoch, den 13. April 2016 in Ihrer Übungsgruppe.

Hinweise: Benutzen Sie ein Array `Bucket[]`, wobei `Bucket[k]` angibt, wie viele verschiedene Arten mit je k Gegenständen verfügbar sind.