

## Algorithmische Anwendungen WS 2003/2004

### Praktikum 3: Dynamische Programmierung zur Optimierung der Matrix-Ketten Multiplikation

Abnahme: 49. u. 50. Woche

Name: ..... Matr-Nr: .....

Datum: ..... Unterschrift des Dozenten (wenn bestanden): .....

**Literatur:** Skript Kapitel 2.3

**Lernziel:** Genaueres Verständnis der dynamischen Programmierung am Beispiel der Optimierung von Matrix-Ketten Multiplikationen.

**Hinweis:** Die in Punkte a)-d) in Aufgabe 1 müssen von jedem Praktikumsteilnehmer gelöst und dokumentiert werden. Die Lösungen sind im Praktikum detailliert zu erklären.

#### Aufgabe 1 ( Verständnis )

Gegeben ist die Matrix-Kette  $\langle A_1, A_2, A_3, A_4 \rangle$  mit dem Dimensions-Vektor  $p = (p_0, p_1, p_2, p_3, p_4) = (5, 3, 4, 2, 20)$ .

a) Schreiben Sie alle **Klammerungen** der Matrix-Kette  $\langle A_1, A_2, A_3, A_4 \rangle$  auf, und prüfen Sie die Anzahl der Klammerungen durch Evaluation der Rekursionsgleichung (11) (siehe Skript Folie 3.2/12)

b) Markieren Sie in jeder Klammerung den Splittpunkt der letzten Multiplikation.

c) Berechnen Sie nach der Rekursionsgleichung (12) (siehe Skript Folie 3.2 / 23) die **Tabelle  $m[i, j]$**  für die Matrixkette  $\langle A_1, A_2, A_3, A_4 \rangle$  und den Dimensionsvektor  $p = (p_0, p_1, p_2, p_3, p_4) = (5, 3, 4, 2, 20)$ . Schreiben Sie jeden Schritt genau auf, und markieren Sie den jeweils ausgewählten min-Wert, der dann in die Tabelle eingetragen wird.

Mit jedem Wert in der  $m$ -Tabelle hat man auch den zugehörigen  $k$ -Wert (des Splittpunktes) gefunden. Tragen Sie den  $k$ -Wert in die **Tabelle  $s[i, j]$**  ein.

Erklären Sie, warum die Tabelle  $s$  nur oberhalb der Nordwest-Südost-Diagonale sinnvolle Werte enthält.

d) Führen Sie einen Trace des Algorithmus *Print-Optimal-Parens*( $s, 1, 4$ ) durch und schreiben Sie die optimale Klammerung auf (siehe Skript Folie 3.2 / 44).

Welcher Zusammenhang besteht zwischen den beim Trace von *Print-Optimal-Parens*( $s, 1, 4$ ) verwendeten  $s$ -Werte (als Parameter für  $i$  und  $j$ ) und der Berechnung der Werte in der  $m$ -Tabelle. **Genauere Erklärung.**

## Aufgabe 2 ( Implementation )

Implementieren Sie die Algorithmen

- *Matrix-Chain-Order* (  $p$  )
- *Print-Optimal-Parens* (  $s, i, j$  )

**Eingabe:** Der Dimensions-Vektor  $p = (p_0, p_1, p_2, \dots, p_n)$  soll interaktiv eingegeben werden. Seine Länge bestimmt somit automatisch die Länge der Matrix-Kette  $\langle A_1, A_2, A_3, \dots, A_n \rangle$ .

Prüfen Sie, ob die durch den Vektor definierten Matrizen kompatibel sind.

**Ausgabe:** Der Algorithmus *Matrix-Chain-Order* (  $p$  ) soll die Tabellen  $m[i, j]$  und  $s[i, j]$  in gut lesbarer Form darstellen.

Der Algorithmus *Print-Optimal-Parens* (  $s, i, j$  ) soll das optimal geklammerte Matrixprodukt darstellen.

**Performance:** Welche asymptotische Laufzeit hat der Algorithmus *Matrix-Chain-Order* (  $p$  ) ?

**Hinweis:** Die Matrix-Multiplikation selbst muss nicht implementiert werden.