

Informatik II

Übung 12

FS 2020

Heutiges Programm

1 Wiederholung Theorie

2 In-Class Exercise

1. Wiederholung Theorie

Dynamische Programmierung: Idee

- Aufteilen eines komplexen Problems in eine vernünftige Anzahl kleinerer Teilprobleme
- Die Lösung der Teilprobleme wird zur Lösung des komplexeren Problems verwendet
- Identische Teilprobleme werden nur einmal gerechnet

Dynamic Programming = Divide-And-Conquer ?

- In beiden Fällen ist das Ursprungsproblem (einfacher) lösbar, indem Lösungen von Teilproblemen herangezogen werden können. Das Problem hat *optimale Substruktur*.
- Bei Divide-And-Conquer Algorithmen (z.B. Mergesort) sind Teilprobleme unabhängig; deren Lösungen werden im Algorithmus nur einmal benötigt.
- Beim DP sind Teilprobleme nicht unabhängig. Das Problem hat *überlappende Teilprobleme*, welche im Algorithmus mehrfach gebraucht werden.
- Damit sie nur einmal gerechnet werden müssen, werden Resultate tabelliert. Dafür darf es *zwischen Teilproblemen keine zirkulären Abhängigkeiten* geben.

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

- **Definition der DP-Tabelle:**

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

- **Definition der DP-Tabelle:** Welche Dimensionen hat die Tabelle? Was ist die Bedeutung jedes Eintrags?

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

- **Definition der DP-Tabelle:** Welche Dimensionen hat die Tabelle? Was ist die Bedeutung jedes Eintrags?
- **Berechnung eines Eintrags:**

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

- **Definition der DP-Tabelle:** Welche Dimensionen hat die Tabelle? Was ist die Bedeutung jedes Eintrags?
- **Berechnung eines Eintrags:** Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

- **Definition der DP-Tabelle:** Welche Dimensionen hat die Tabelle? Was ist die Bedeutung jedes Eintrags?
- **Berechnung eines Eintrags:** Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?
- **Berechnungsreihenfolge:**

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

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

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

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

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

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

Dynamische Programmierung

Eine vollständige Beschreibung eines dynamischen Programms behandelt **immer** die folgenden Aspekte :

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

2. In-Class Exercise

Longest Ascending Sequence in a Matrix

Längste aufsteigende Sequenz in Matrix

Gegeben $n \times m$ Matrix A :

9	27	42	41	48
35	39	8	3	5
12	49	2	38	4
15	47	29	28	6
19	1	25	33	10

Längste aufsteigende Sequenz in Matrix

Gegeben $n \times m$ Matrix A :

9	27	42	41	48
35	39	8	3	5
12	49	2	38	4
15	47	29	28	6
19	1	25	33	10

Gesucht längste aufsteigende Sequenz:

4, 6, 28, 29, 47, 49

Definition der DP-Tabelle

- Welche Dimensionen hat die Tabelle?

Definition der DP-Tabelle

- Welche Dimensionen hat die Tabelle?
 - $n \times m$

Definition der DP-Tabelle

- Welche Dimensionen hat die Tabelle?
 - $n \times m(\times 2)$

Definition der DP-Tabelle

- Welche Dimensionen hat die Tabelle?
 - $n \times m(\times 2)$
- Was ist die Bedeutung jedes Eintrags?

Definition der DP-Tabelle

- Welche Dimensionen hat die Tabelle?
 - $n \times m(\times 2)$
- Was ist die Bedeutung jedes Eintrags?
 - In $T[x][y]$ steht Länge der längsten aufsteigenden Sequenz, die im Feld $A[x][y]$ endet
 - In $S[x][y]$ steht Koordinaten des Vorgängers von (x, y) in aufsteigender Sequenz (falls existent)

Berechnung eines Eintrags

- Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?

Berechnung eines Eintrags

- Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?
 - Betrachte Nachbarn mit kleineren Eintrag in A .

Berechnung eines Eintrags

- Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?
 - Betrachte Nachbarn mit kleineren Eintrag in A .
 - Wähle von den kleineren Einträgen den mit dem grössten Eintrag in T

Berechnung eines Eintrags

- Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?
 - Betrachte Nachbarn mit kleineren Eintrag in A .
 - Wähle von den kleineren Einträgen den mit dem grössten Eintrag in T
 - Aktualisiere T und S . (S erhält Koordinaten vom ausgewählten Nachbar, T erhält Wert um eins erhöht vom ausgewählten Nachbar.)

Berechnung eines Eintrags

- Wie berechnet sich ein Eintrag aus den Werten von anderen Einträgen? Welche Einträge hängen nicht von anderen Einträgen ab?
 - Betrachte Nachbarn mit kleineren Eintrag in A .
 - Wähle von den kleineren Einträgen den mit dem grössten Eintrag in T
 - Aktualisiere T und S . (S erhält Koordinaten vom ausgewählten Nachbar, T erhält Wert um eins erhöht vom ausgewählten Nachbar.)

Berechnungsreihenfolge

- In welcher Reihenfolge kann man die Einträge berechnen, so dass die jeweils benötigten anderen Einträge bereits vorher berechnet wurden?

Berechnungsreihenfolge

- In welcher Reihenfolge kann man die Einträge berechnen, so dass die jeweils benötigten anderen Einträge bereits vorher berechnet wurden?
- Bottom-Up: Beginne mit kleinstem Element in A und so weiter. (Bedeutet dass man A sortieren muss.)

Berechnungsreihenfolge

- In welcher Reihenfolge kann man die Einträge berechnen, so dass die jeweils benötigten anderen Einträge bereits vorher berechnet wurden?
- Bottom-Up: Beginne mit kleinstem Element in A und so weiter. (Bedeutet dass man A sortieren muss.)
- Rekursiv: Beliebige Reihenfolge, falls Eintrag schon berechnet überspringen sonst rekursiv von kleinern Nachbarn berechnen.

Auslesen der Lösung

- Wie lässt sich die Lösung am Ende aus der Tabelle auslesen?

Auslesen der Lösung

- Wie lässt sich die Lösung am Ende aus der Tabelle auslesen?
 - Betrachte alle Einträge um den Eintrag zu finden, in dem eine längste Sequenz endet. Von dort aus können wir die Lösung rekonstruieren, indem wir dem entsprechenden Vorgänger folgen.

Aufgabe

Implementieren Sie eine Lösung mit Dynamic Programming im Code Expert Programm ()

Fragen oder Anregungen?