Föreläsning

Mer dynamisk programmering

- Hur känner man igen problem som kan lösas med dyn. prog.?
  1. Ska ha optimala delstrukturer, dvs lösningen till (del)problem kan uttryckas rekursivt
  2. Ska ha överlappande delproblem – annars är dekomposition bättre
- Hur kan man ställa upp rekursionen?
- Hur visar man korrekthet?
- Memoisering (memoization)

Alternativ implementation av rekursionen:
Sist i funktionen sparas funktionsvärdet, och först i funktionen kollas om det finns sparat värde.
Är inbyggt i vissa språk.
Tar något längre tid pga sämre minneslokalitet

1. Optimala lösningens struktur

Stig uppbyggd av delstigar

2. Rekursion

Låt $V[i,j]$ = värdet på bästa stigen från a_{ij} ner till rad n.

$$V[i,j] = \begin{cases} a_{ij} & \text{om } i=n \text{ (dvs sista raden)} \\ a_{ij} + \max(V[i+1,j], V[i+1,j+1]) & \text{annars} \end{cases}$$

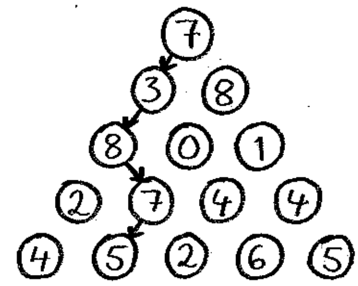
3. Beräkning

```
for j ← 1 to n do
    V[n,j] ← a_nj
for i ← n-1 downto 1 do
    for j ← 1 to i do
        V[i,j] ← a_ij + max(V[i+1,j], V[i+1,j+1])
return V[1,1]
```

Tidskomplexitet: $O(n^2)$

Exempel 1: Maximal triangelstig

Problem: Hitta stigen från toppen ner till botten som maximerar summan av dom ingående talen.



n = antal rader

Finns 2^{n-1} stigar att pröva

a_{ij} = element nr j på rad i

VISA KORREKTHET FÖR DYNPROGALGORITM

REKURSIONSRELATIONEN GÖR KORREKTHETS-BEVISET ENKLARE!

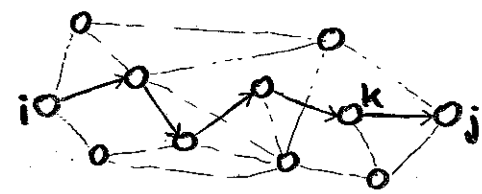
- VISA ATT REKURSIONEN BESKRIVER LÖSNINGEN TILL PROBLEMET — NORMALT ENKELT, KRÄVER INGEN INBLANDNING AV PROGRAMKOD

- VISA ATT PROGRAMKODEN BERÄKNAR REKURSIONEN KORREKT.
  - VISA ATT BASFALLEN BERÄKNAS KORREKT
  - VISA ATT BERÄKNINGSORDNINGEN ÄR OKEJ, DVS VÄRDEN BERÄKNAS INNAN DOM ANVÄNDS
  - VISA ATT REKURSIONSSTEGET BERÄKNAS KORREKT

EXEMPEL: KORTASTE STIG MELLAN ALLA PAR AV HÖRN I GRAF MED KANTVIKTER

- HITTA OPTIMALA DELSTRUKTURER!

IDÉ: VARJE DELSTIG TILL EN OPTIMAL STIG ÄR OPTIMAL



OM KORTASTE STIGEN FRÅN HÖRN **i** TILL HÖRN **j** BESTÅR AV m KANTER DÄR DEN SISTA GÅR FRÅN HÖRN **k** TILL **i**
SÅ BESTÅR KORTASTE STIGEN FRÅN **i** TILL **k** AV $m-1$ KANTER OCH

$$\delta(i,j) = \delta(i,k) + w_{kj}$$

DÄR $\delta(i,j)$ = LÄNGDEN AV KORTASTE STIGEN FRÅN i TILL j
w_{ij} = VIKTEN PÅ KANTEN FRÅN i TILL j

- DEFINIERA OPTIMALA LÖSNINGENS VÄRDE REKURSIVT

LÅT $d_{ij}^{(m)}$ = LÄNGDEN AV DEN KORTASTE STIGEN FRÅN i TILL j SOM INNEHÅLLER $\le m$ KANTER
$d_{ij}^{(m)} = \infty$ OM INGEN STIG MED $\le m$ KANTER FINNS

ANTA ATT $w_{jj} = 0$
$w_{ij} = \infty$ OM INGEN KANT FINNS MELLAN i OCH j.

$$d_{ij}^{(1)} = w_{ij}$$

$$d_{ij}^{(m)} = \min\left(d_{ij}^{(m-1)}, \min_{\substack{1\le k\le n\\ k\ne j}}\{d_{ik}^{(m-1)} + w_{kj}\}\right) = \min_{1\le k\le n}\{d_{ik}^{(m-1)} + w_{kj}\}$$

$$\delta(i,j) = d_{ij}^{(n-1)} \left(= d_{ij}^{(n)} = d_{ij}^{(n+1)} = \dots\right)$$

- BERÄKNA $d_{ij}^{(m)}$ FRÅN SMÅ m TILL STORA

- $D \leftarrow (w_{ij})$
- FOR $m \leftarrow 2$ TO $n-1$ DO
  $D \leftarrow$ EXTEND(D, W)
- RETURN D

($D \leftarrow (w_{ij})$
FOR $i \leftarrow 1$ TO $\log n$ DO
$D \leftarrow$ EXTEND(D, D)
RETURN D)

EXTEND(D, W) = FOR $i \leftarrow 1$ TO n DO
FOR $j \leftarrow 1$ TO n DO
$d'_{ij} \leftarrow \infty$
FOR $k \leftarrow 1$ TO n DO
$d'_{ij} \leftarrow$ MIN $(d'_{ij}, d_{ik} + w_{kj})$
RETURN (d'_{ij})

$O(n^3)$