Dinamik Programlama

Tanım = Böl ve Yönet'e karşı olarak çıkarılmıştır

→ Böl ve Yönet yaklaşımı ortak alt programları tekrar tekrar çözecektir.

Dinamik Programlama her alt problemi bir kez çözer ve cevabı bir tabloda saklar.

→ Matris Zincir Çarpımı formülü kullanılır

$$A_{i \times k} \times B_{k \times j} = \underline{\underline{C_{i \times j}}}$$

↓

elde etmek için toplam çarpım sayısı

$i \times k \times j$ = kadar

Çarpım Sırası

$A_1 : 10 \times 100$

$A_2 : 100 \times 5$

$A_3 : 5 \times 50$

1-) $(A_1 . A_2) . A_3 \rightarrow A_1 . A_2 = 10 \times 100 \times 5 = 5000$ → adet çarpma

$A_{12} = 10 \times 5$

$A_{12} . A_3 = 10 \times 5 \times 50 = 2500$ → adet çarpma

$A_{123} = \underline{10 \times 50}$ → 7500 skalar çarpım

matris boyutu

2-) $A_1 . (A_2 . A_3) \rightarrow A_2 . A_3 = 100 \times 5 \times 50 = 25000$ → adet çarpım

$A_{23} = 100 \times 50$

$A_1 . A_{23} = 10 \times 100 \times 50 = 50000$ → adet çarpım

$A_{123} = \underline{10 \times 50}$ → 75.000 skaler çarpım

matris boyutu

Bu bir optimizasyon problemi.

! Min sayıda çarpım yapabilmek için matrisler hangi sıra ile çarpmalı

Olay Parantezleme

$$(A_1 \cdots A_k) \cdot (A_{k+1} \cdots A_n)$$

$$\sum_{k} f(k) \cdot f(n-k)$$

farklı biçimde

farklı biçimde

$f(n) = ?$

$$f(n) = \frac{1}{n} \cdot \binom{2n-2}{n-1}$$

farklı çarpım sayısıdır

$n=4$ için

$$\frac{1}{4} \cdot \binom{6}{3} = \frac{1}{4} \cdot \frac{6 \cdot 5 \cdot 4}{3 \cdot 2 \cdot 1} = 5$$

$n=10$ için

$$\frac{1}{10} \cdot \binom{18}{9} = \frac{1}{10} \cdot \frac{18 \cdot 17 \cdot 16 \cdot 15 \cdot 14 \cdot 13 \cdot 12 \cdot 11 \cdot 10}{9 \cdot 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 \cdot 3 \cdot 2}$$

$= 4862$ farklı çarpım sırası vardır

$n = 1 \ldots$

1, 1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796
1 2 3 4 5 6 7 8 9 10 11 = n

Optimal Parantezleme

$A_{i \ldots j} = \underbrace{A_{i \ldots k} \cdot A_{k+1 \ldots j}}$ → optimal parantezleme olur

↳ A_i matrisinden A_j matrisine kadar çarpma

$1 \leq i \leq j \leq n$

$m[i, j] = A_{i \ldots j}$'yi hesaplamak için gereken min

çarpma sayısını ifade eder

Problem = $A_{1 \ldots n}$ i $m[1, n]$

$A_{i \ldots j} = A_{i \ldots k} A_{k+1 \ldots j}$

Ana Formül

$m[i\,j] = m[i\,k] + m[k+1, j] + P_{i-1} P_k P_j$

$$
m[i,j] = \begin{cases} 0 & \text{if } i=j \\ \min\limits_{i \le k < j} \{ m[i,k] + m[k+1, j] + P_{i-1} P_k P_j & \text{if } i<j \} \end{cases}
$$

Ör

$A_1 . A_2 . A_3$ Soru

	1	2	3
3	7500 $k=2$	25000 $k=2$	0
2	5000 $k=1$	0	
1	0		

A_1: 10×100 $(P_0 \times P_1)$

A_2: 100×5 $(P_1 \times P_2)$

A_3: 5×50 $(P_2 \times P_3)$

5000

$$m\overset{i\ \ j}{[1,2]} = m[1,1] + m[2,2] + P_{i-1} P_k P_2$$

$k=1$ 0 0 P_0 P_1 P_1 10 100 5

25000

$$m\overset{i\ \ j}{[2,3]} = m[2,2] + m[3,3] + P_1 P_2 P_3$$

$k=2$ 0 0 100 5 50

25000 + 50000 = 75000

$$m\overset{i\ \ j}{[1,3]} = m[1,1] + m[2,3] + P_0 P_1 P_3$$

$k=1$ 0 25000   10 100 50

$k=2 \rightarrow m[1,2] + m[3,3] + P_0 P_2 P_3$ 2500 + 5000 = 7500

5000   0   10 . 5 . 50

$A_1 A_2 A_3 = (A_1 . A_2) . A_3$

tabloda

$k=2$ yazılmış

j \ i	1	2	3	4	5	6
6	3	3	3	5	5	0
5	3	3	3	4	0	
4	3	3	3	0		
3	1	2	0			
2	1	0				
1	0					

$s[i,j]$ = $A_i A_{i+1} \cdots A_j$ zincirini A_k ve A_{k+1} arasında parantezle ayıran k değeri

$$A_{1-6} = (A_{1\cdots 3})\cdot(A_{4\cdots 6})$$

$$= (A_1\cdot(A_2\cdot A_3))\cdot((A_4\cdot A_5)\cdot A_6) \rightarrow \text{en optimal çarpma şekli}$$

Maliyet → $\Theta(n^2)$ kadar alt problem var.

Zaman → $\Theta(n^3)$'lük bir hesap maliyeti

Bellek → $\Theta(n^2)$ bellek maliyeti