Dynamic Programming
Chain Matrix Multiply

(Class 20)

- Let $m[i, j]$ denote the minimum number of multiplications needed to compute $A_{i \dots j}$.
- Where $1 \leq i \leq j \leq n$ (i and j are number of matrices).
- The optimum can be described by the following recursive formulation.
  - If $i = j$, there is only one matrix and thus $m[i, i] = 0$ (the diagonal entries).
  - If $i < j$, we are asking for the product $A_{i \dots j}$.
  - This can be split by considering each k, $i \leq k < j$, as $A_{i \dots k}$ times $A_{k+1 \dots j}$.

- The optimum time to compute $A_{i\ldots k}$ is $m[i,k]$ and optimum time for $A_{k+1\ldots j}$ is in $m[k+1,j]$.
- Since $A_{i\ldots k}$ is a $p_{i-1} \times p_k$ matrix and $A_{k+1\ldots j}$ is $p_k \times p_j$ matrix, the time to multiply them is $p_{i-1} \times p_k \times p_j$.
- This suggests the following recursive rule:

$$m[i,i] = 0$$
$$m[i,j] = \min_{i \le k < j} (m[i,k] + m[k+1,j] + p_{i-1}p_kp_j)$$

- We do not want to calculate m entries recursively.
- So how should we proceed? We will fill m along the diagonals.
- Set all $m[i, i] = 0$ using the base condition.
- Compute cost for multiplication of a sequence of 2 matrices.

- These are $m[1,2], m[2,3], m[3,4], \ldots , m[n-1, n]$.
- For example, $m[1,2]$ is:

$$m[1,2] = m[1,1] + m[2, 2] + p_0 \cdot p_1 \cdot p_2$$

- For example, the m for product of 5 matrices at this stage would be:

$m[1,1]$	$\leftarrow m[1,2]$ $\downarrow$			
	$m[2,2]$	$\leftarrow m[2,3]$ $\downarrow$		
		$m[3,3]$	$\leftarrow m[3,4]$ $\downarrow$	
			$m[4,4]$	$\leftarrow m[4,5]$ $\downarrow$
				$m[5,5]$

- Next, we compute cost of multiplication for sequences of three matrices.
- These are $m[1,3], m[2,4], m[3,5], \ldots, m[n-2,n]$.
- For example, $m[1,3]$ is:

$$m[1,3] = \min \begin{cases} m[1,1] + m[2,3] + p_0 \cdot p_1 \cdot p_3 \\ m[1,2] + m[3,3] + p_0 \cdot p_2 \cdot p_3 \end{cases}$$

- We repeat the process for sequences of four, five and higher number of matrices.
- The final result will end up in $m[1,n]$.

- **Example**: We want to find the optimal multiplication order for:

$$\underset{(5\times4)}{A_1} \cdot \underset{(4\times6)}{A_2} \cdot \underset{(6\times2)}{A_3} \cdot \underset{(2\times7)}{A_4} \cdot \underset{(7\times3)}{A_5}$$

- We will compute the entries of the m matrix starting with the base condition.
- We first fill that main diagonal using base case:

0				
	0			
		0		
			0	
				0

- Next, we compute the entries in the first super diagonal, i.e., the diagonal above the main diagonal:

$$m[1,2] = m[1,1] + m[2,2] + p_0 \cdot p_1 \cdot p_2 = 0 + 0 + (5 \cdot 4 \cdot 6) = 120$$

$$m[2,3] = m[2,2] + m[3,3] + p_1 \cdot p_2 \cdot p_3 = 0 + 0 + (4 \cdot 6 \cdot 2) = 48$$

$$m[3,4] = m[3,3] + m[4,4] + p_2 \cdot p_3 \cdot p_4 = 0 + 0 + (6 \cdot 2 \cdot 7) = 84$$

$$m[4,5] = m[4,4] + m[5,5] + p_3 \cdot p_4 \cdot p_5 = 0 + 0 + (2 \cdot 7 \cdot 3) = 42$$

- The matrix m now looks as follows after the product of two matrices (first super diagonal):

0	120			
	0	48		
		0	84	
			0	42
				0

- We now proceed to the second super diagonal.
- This time, however, we will need to try two possible values for k.
- For example, there are two possible splits for computing $m[1, 3]$.

- We will choose the split that yields the minimum:

$$m[1,3] = m[1,1] + m[2,3] + p_0 \cdot p_1 \cdot p_3 = 0 + 48 + (5 \cdot 4 \cdot 2) = 88$$

$$m[1,3] = m[1,2] + m[3,3] + p_0 \cdot p_2 \cdot p_3 = 120 + 0 + (5 \cdot 6 \cdot 2) = 180$$

$$the\ minimum\ m[1,3] = 88\ occurs\ with\ k = 1$$

- Similarly, for $m[2, 4]$ and $m[3, 5]$:

$$m[2,4] = m[2,2] + m[3,4] + p_1 \cdot p_2 \cdot p_4 = 0 + 48 + (4 \cdot 6 \cdot 7) = 252$$

$$m[2,4] = m[2,3] + m[4,4] + p_1 \cdot p_3 \cdot p_4 = 48 + 0 + (4 \cdot 2 \cdot 7) = 104$$

$$the\ minimum\ m[2,4] = 104\ at\ k = 3$$

$$m[3,5] = m[3,3] + m[4,5] + p_2 \cdot p_3 \cdot p_5 = 0 + 42 + (6 \cdot 2 \cdot 3) = 78$$

$$m[3,5] = m[3,4] + m[5,5] + p_2 \cdot p_4 \cdot p_5 = 84 + 0 + (6 \cdot 7 \cdot 3) = 210$$

$$the\ minimum\ m[3,5] = 78\ at\ k = 3$$

- With the second super diagonal computed, the m matrix looks as follow:

0	120	88		
	0	48	104	
		0	84	78
			0	42
				0

- We repeat the process for the remaining diagonals.
- However, the number of possible splits (values of k) increases:

$$m[1,4] = m[1,1] + m[2,4] + p_0 \cdot p_1 \cdot p_4 = 0 + 104 + (5 \cdot 4 \cdot 7) = 244$$

$$m[1,4] = m[1,2] + m[3,4] + p_0 \cdot p_2 \cdot p_4 = 120 + 84 + (5 \cdot 6 \cdot 7) = 414$$

$$m[1,4] = m[1,3] + m[4,4] + p_0 \cdot p_3 \cdot p_4 = 88 + 0 + (5 \cdot 2 \cdot 7) = 158$$

$$the\ minimum\ m[1,4] = 158\ at\ k = 3$$

$$m[2,5] = m[2,2] + m[3,5] + p_1 \cdot p_2 \cdot p_5 = 0 + 78 + (4 \cdot 6 \cdot 3) = 150$$

$$m[2,5] = m[2,3] + m[4,5] + p_1 \cdot p_3 \cdot p_5 = 48 + 42 + (4 \cdot 2 \cdot 3) = 114$$

$$m[2,5] = m[2,4] + m[5,5] + p_1 \cdot p_4 \cdot p_5 = 104 + 0 + (4 \cdot 7 \cdot 3) = 188$$

$$the\ minimum\ m[2,5] = 114\ at\ k = 3$$

- The matrix m at this stage is:

0	120	88	158	
	0	48	104	114
		0	84	78
			0	42
				0

That leaves the $m[1, 5]$ which can now be computed:

$$m[1,5] = m[1,1] + m[2,5] + p_0 \cdot p_1 \cdot p_5 = 0 + 114 + (5 \cdot 4 \cdot 3) = 174$$
$$m[1,5] = m[1,2] + m[3,5] + p_0 \cdot p_2 \cdot p_5 = 120 + 78 + (5 \cdot 6 \cdot 3) = 288$$
$$m[1,5] = m[1,3] + m[4,5] + p_0 \cdot p_3 \cdot p_5 = 88 + 42 + (5 \cdot 2 \cdot 3) = 160$$
$$m[1,5] = m[1,4] + m[5,5] + p_0 \cdot p_4 \cdot p_5 = 158 + 0 + (5 \cdot 7 \cdot 3) = 263$$
$$the\ minimum\ m[1,5] = 160\ at\ k = 3$$

- We thus have the final cost matrix m as:

0	120	88	158	***160***
0	0	48	104	114
0	0	0	84	78
0	0	0	0	42
0	0	0	0	0

- Here is the order in which m entries are calculated:

0	1	5	8	10
0	0	2	6	9
0	0	0	3	7
0	0	0	0	4
0	0	0	0	0

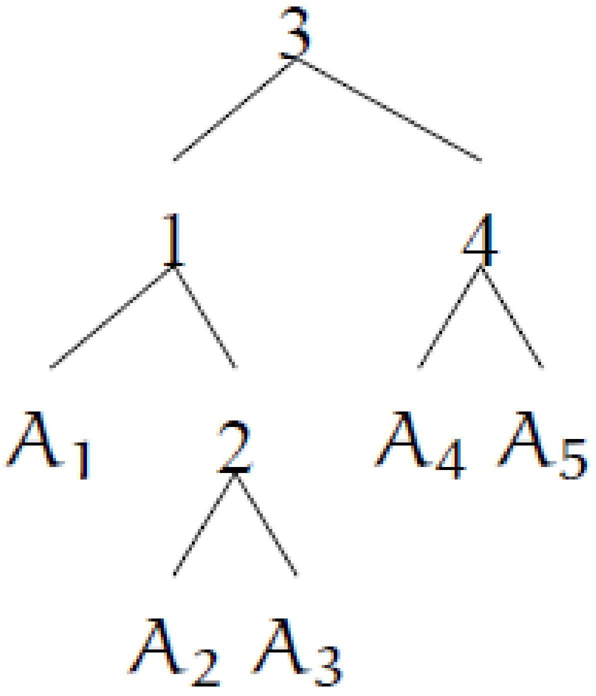
- And the split k values that led to a minimum $m[i, j]$ value:

0	1	1	3	3
	0	2	3	3
		0	3	3
			0	4
				0

- Based on the computation, the minimum cost for multiplying the five matrices is 160 and the optimal.
- So, the optimal order for multiplication is:

$$((A_1(A_2A_3))(A_4A_5))$$

- This can be represented as a binary tree:



Optimum matrix multiplication order for the five matrices example