25.09.2024

Matrix Multiplication :

$$A \times B \times C$$

10×5 $\quad$ 5×20 $\quad$ 20×50.

10×20 $\quad$ 20×50.

$(A \times B) \times C$

$10 \times 5 \times 20$.

$= 1000$

$+ \; 10 \times 20 \times 50$

$= 10000$

10×5 $\quad$ 5×50

$A \times (B \times C)$

$5 \times 20 \times 50$.

$= 5000$

$+ \; 10 \times 5 \times 50$.

$= 2500$

7500

$$\underbrace{A_1}_{m_0 \times m_1} \times \underbrace{A_2}_{m_1 \times m_2} \times \cdots)(\cdots \times \underbrace{A_n}_{m_{n-1} \times m_n}.$$

- greedy? perform the cheapest consecutive pair first
  - this does not work.
- divide & conquer: - will also fail
- DP: DP Table?
  what is the good way to define subproblems?

$$\underbrace{(A_1 \times A_2 \times \cdots A_i)}\times(A_{i+1} \qquad \times A_n.)$$

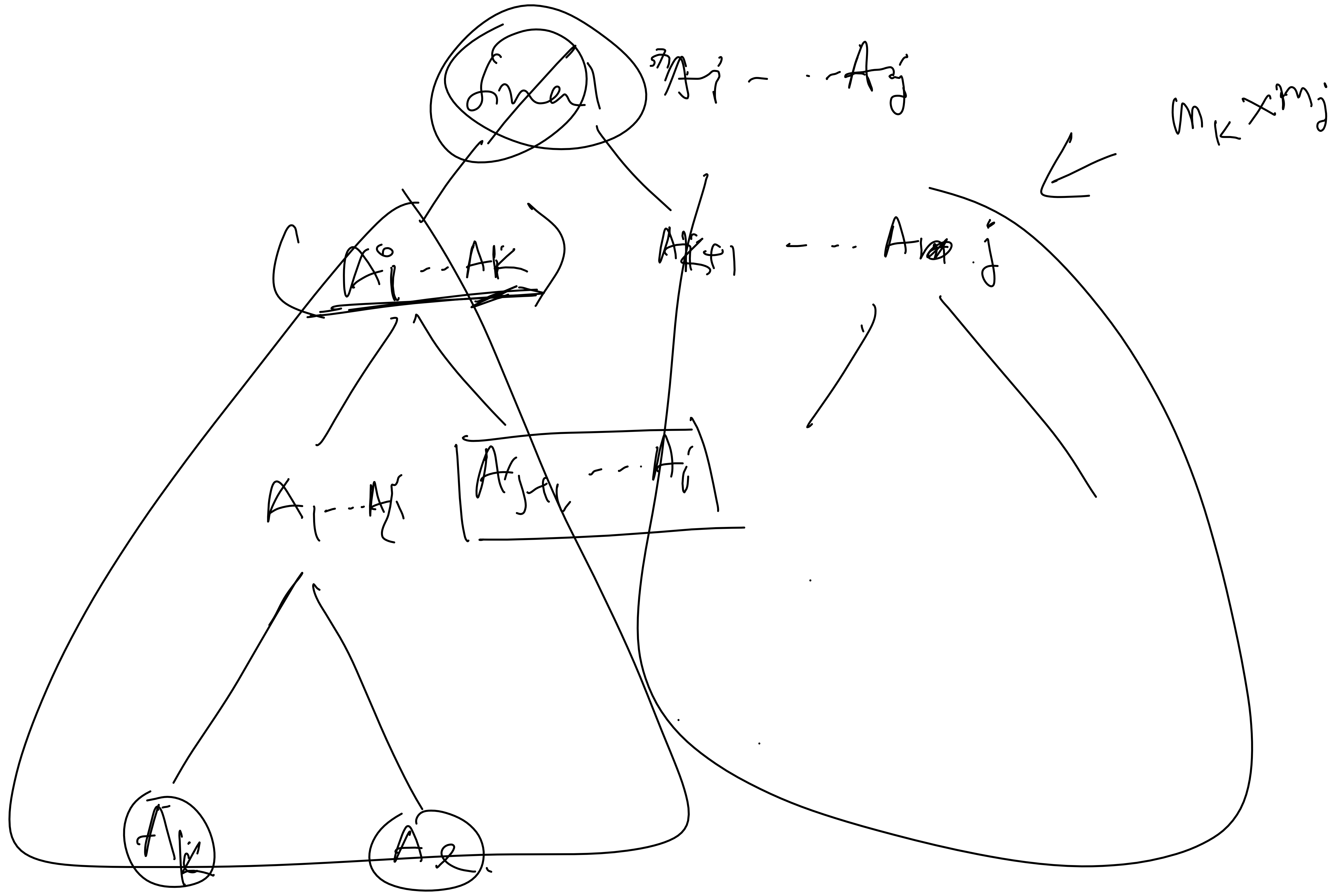
T_n : # ways you can perform

$$T_n = \sum_{i=1}^{n-1} T_i \cdot T_{n-i} \longleftarrow \text{Catalan Numbers.}$$

$$= 2^{\theta(n)}$$

$$\underline{A_1 \times A_2 \times A_3}$$
$$\downarrow$$

$$\underline{A_1 \times A_2 \times A_3 \times A_4}$$
$$\downarrow$$



$$(m_{i-1} \times m_i) \cdots (m_{k-1} \times m_k) = \underline{(m_{i-1} \times m_k)}$$

$M[i,j] \leftarrow$ min # operations needed to

$A_i \times \cdots \times A_j \qquad j \geq i$

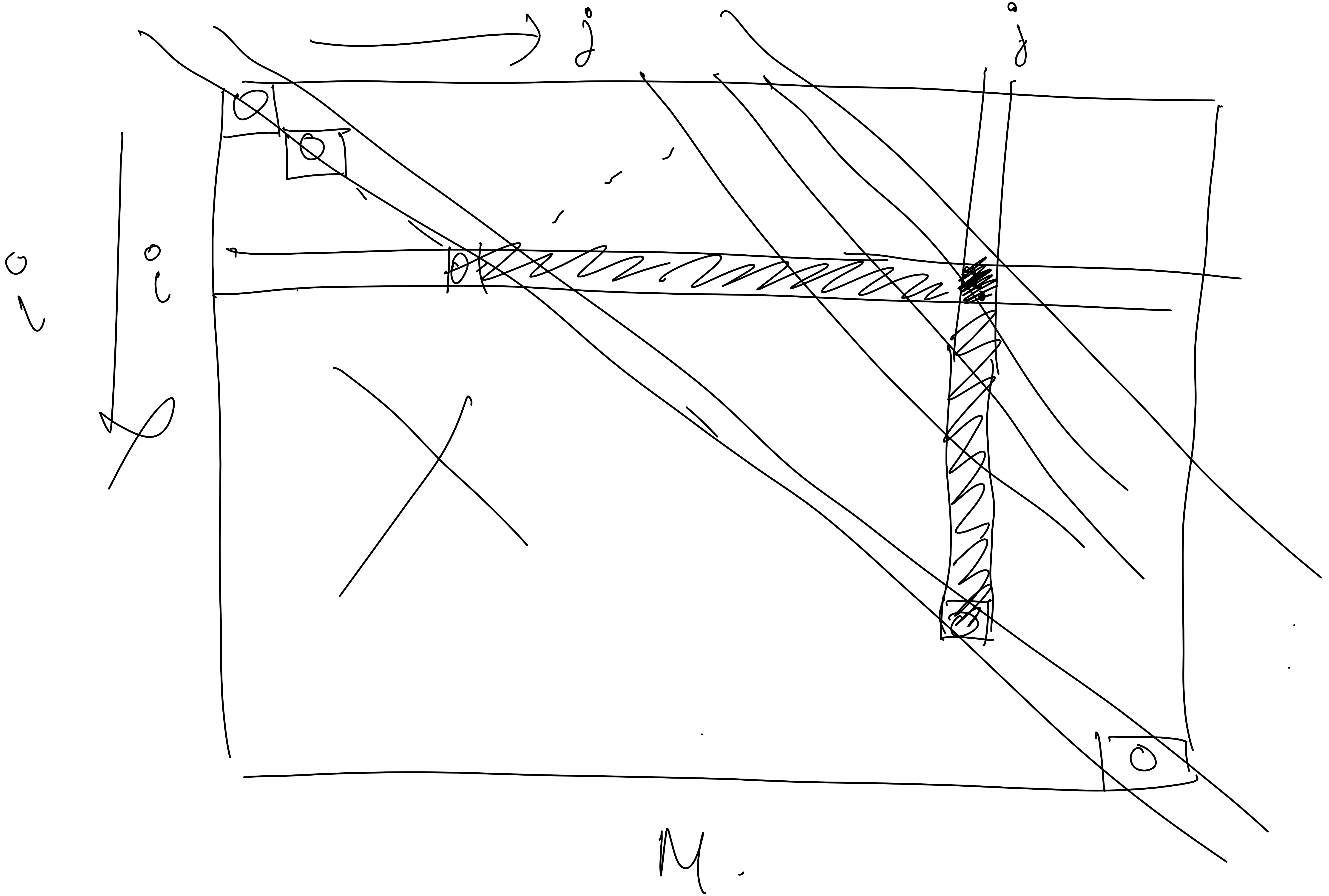
$M[i,i] \leftarrow 0 \quad \forall i$

$M[1,n] \leftarrow$ final answer.

what is the recursive step?

$$M[i,j] \leftarrow \min_{i \le k < j} \left\{ \boxed{M[i,k]} + \boxed{M[k+1,j]} + m_{i-1} \cdot m_k \cdot m_j \right.$$

$$i < j$$


j
j
i
i
M.

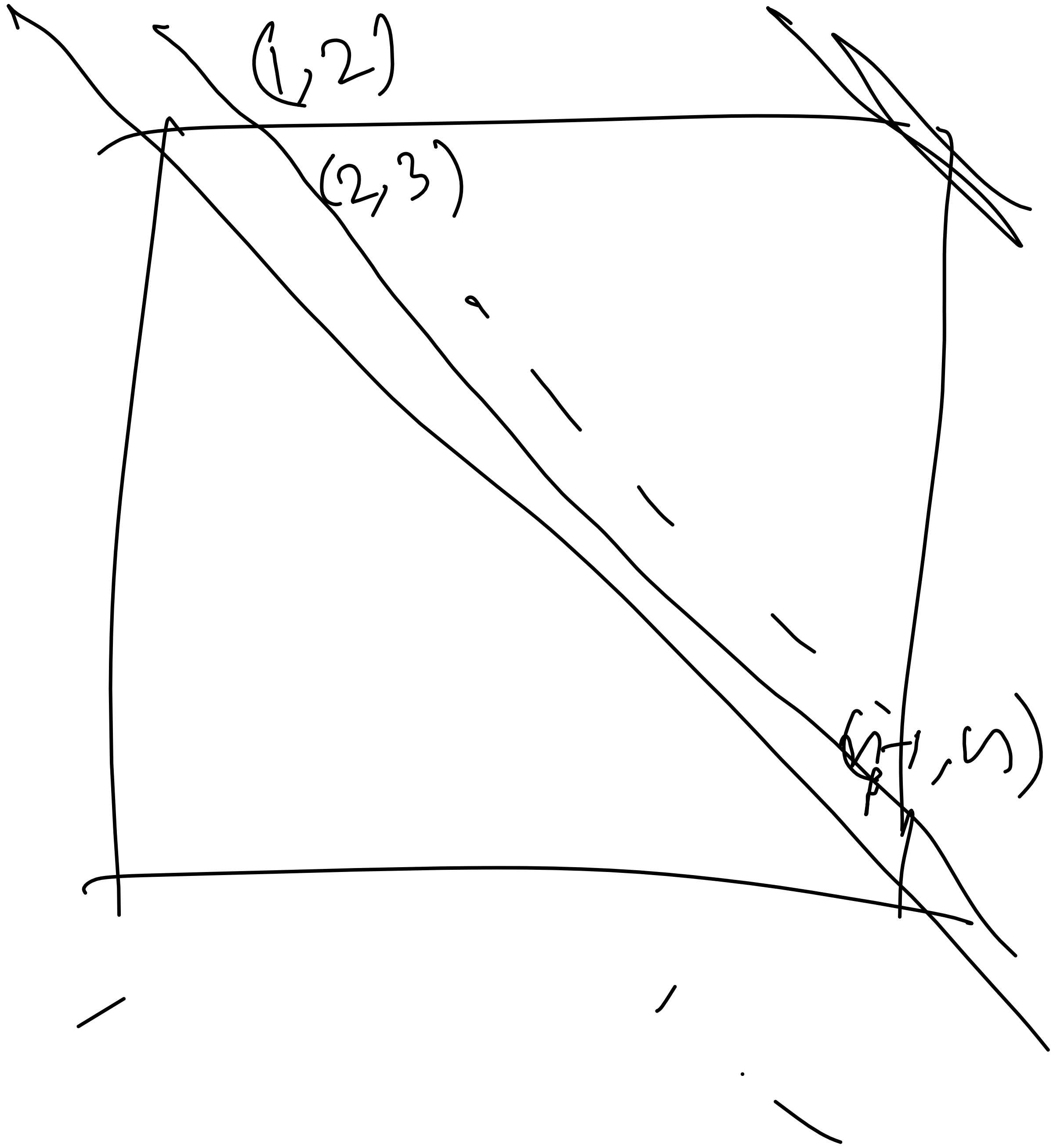
j-th diagonal:

$(1, j)$

$(2, j+1)$

$\vdots$

$(n-j+1, n)$



Procedure MinMult $(m_0, \ldots, m_n)$

for $j = 1$ to n

for $i = 1$ to $(n-j+1)$

if $(j = 1)$ $M[i, i] \leftarrow 0$

else

$M[i, i+j-1] = \min\limits_{i \le k < i+j-1} \left\{ M[i, k] + M[k+1, i+j-1] + m_{i-1} \cdot m_k \cdot m_{i+j-1} \right.$

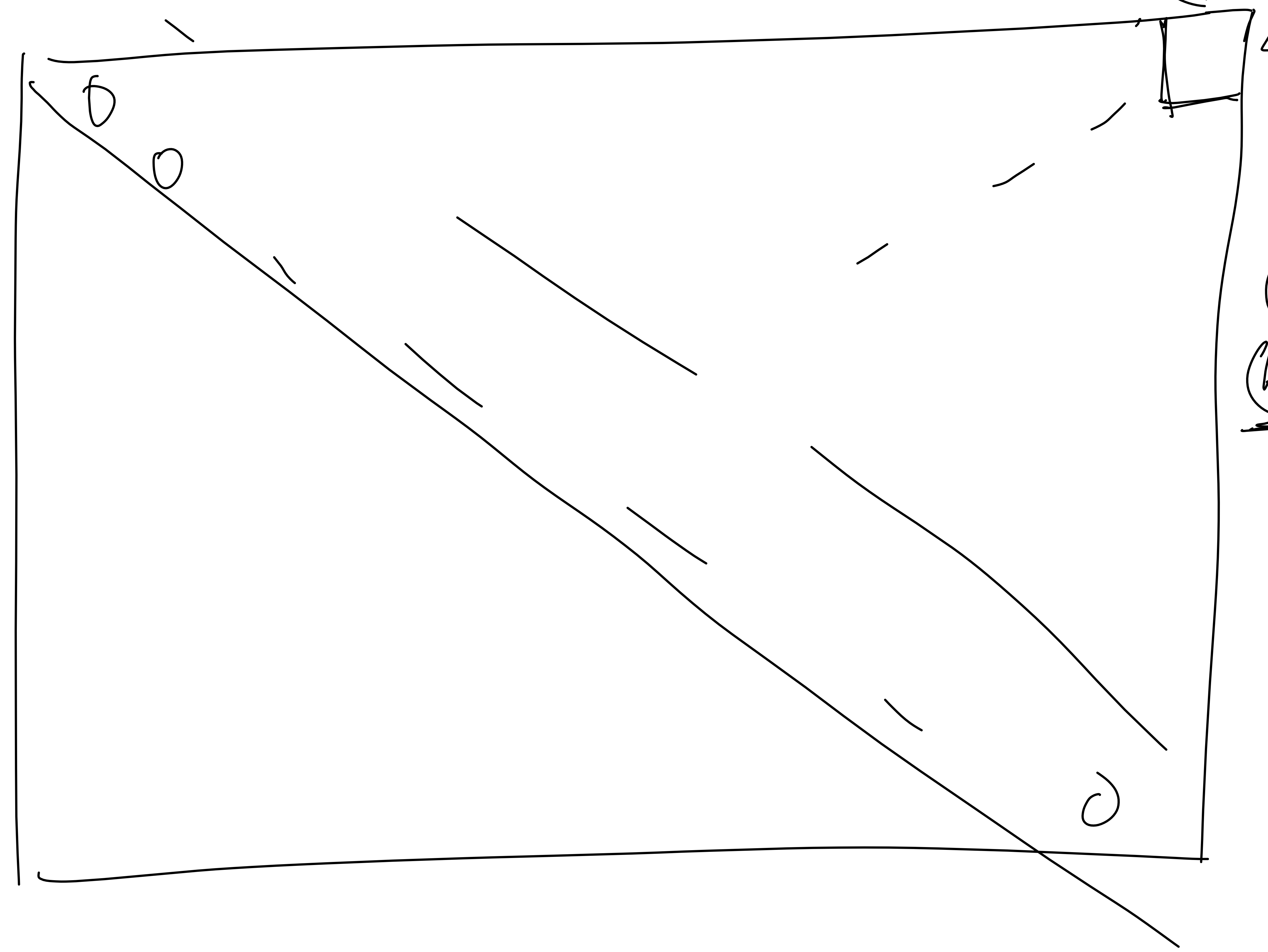
ith entry of jth diagonal.

$A_i \times \ldots \mid \ldots \times A_{i+j-1}$

$k \mid k+1$

$\text{argmin}[i, i+j-1] \leftarrow k^*$

(1,n)



final answer.

(1,k) (k+1,n)

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

All Pair Shortest Path.

(1) edge weight can be negative

(2) source vertex is all possible.

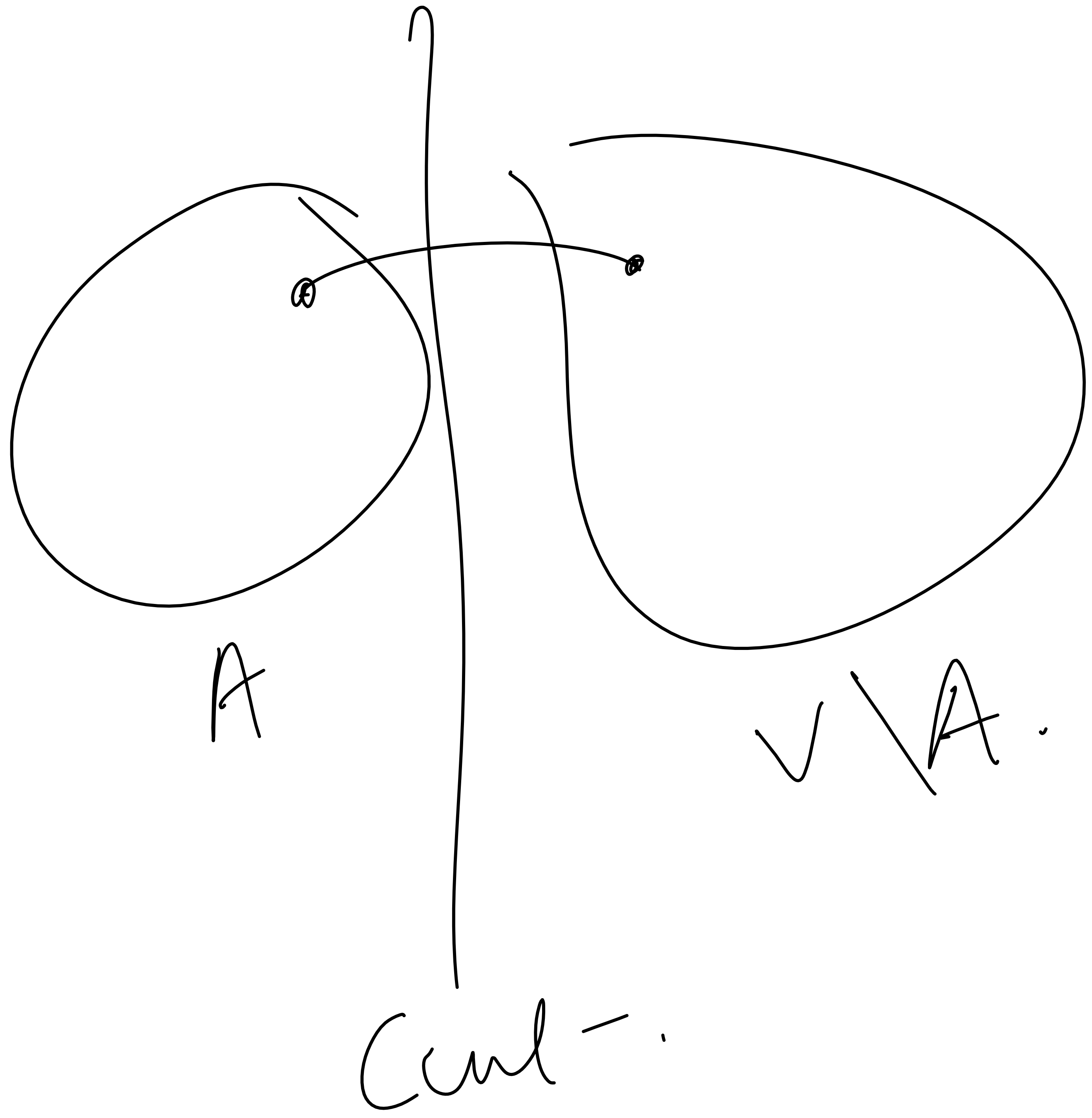
this was fixed in Dijkstra.

Assume edge weight are +ve.

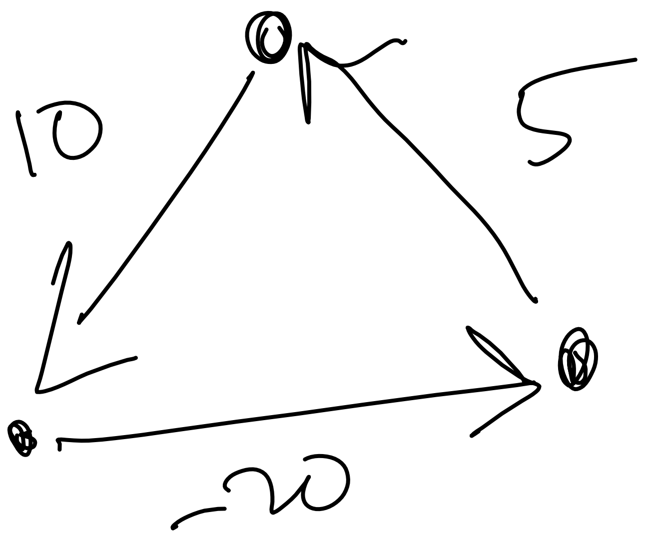
$$O((m+n)\log n \cdot n)$$

$$O(n^3).$$

$m \approx n^2$.


A
V\A.
Curl-.

Assumption: $\nexists$ any $-$ve weight cycles.



ill-defined.

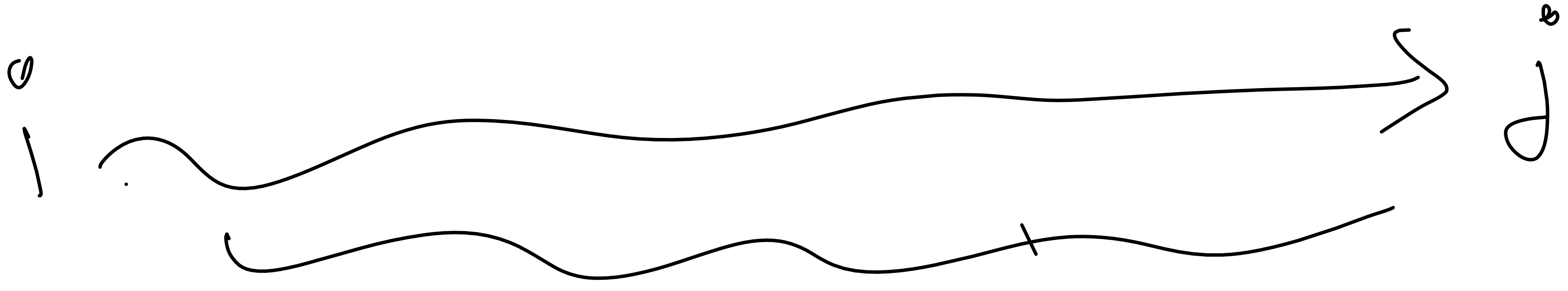
$-\infty$ cost.

We'll see a D.P solution.

$dist(i, j, k)$

vertices: $\{1, 2, \ldots, n\}$

shortest path distance from vertex i to vertex j



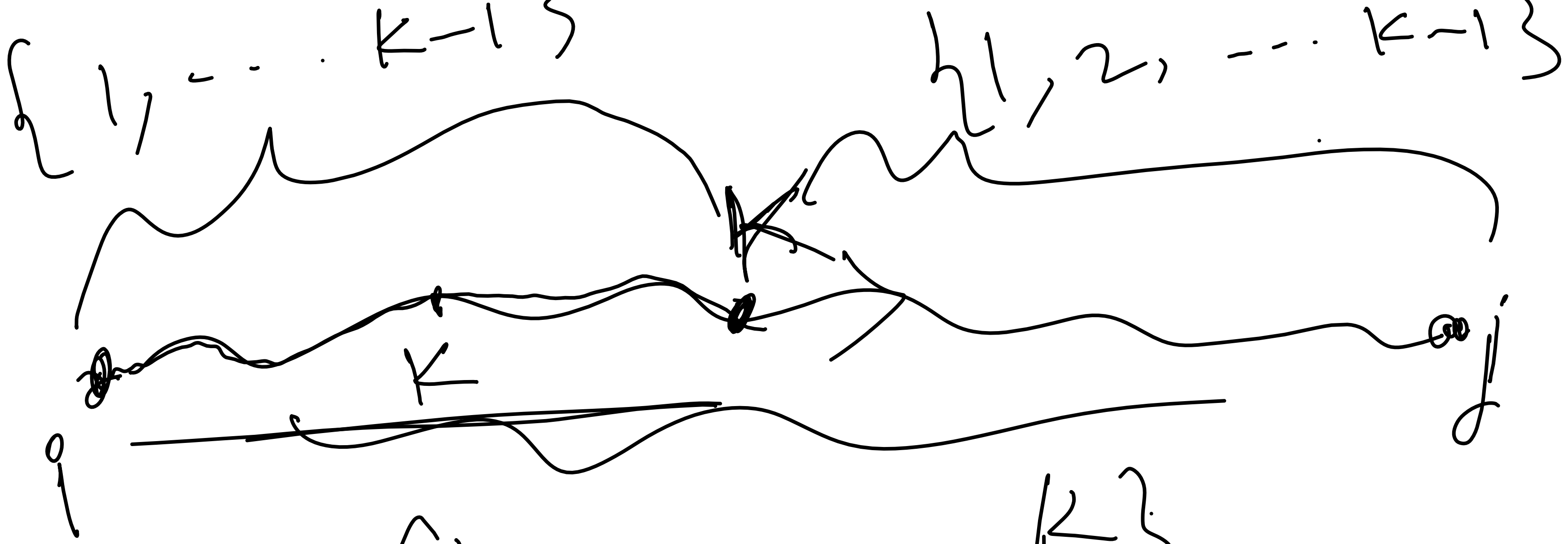
restrict. intermediate vertices.

$\{1, 2, \ldots, k\}$

$dist(i, j, 0) \leftarrow wt(i, j)$ if $(i, j) \in E$

$\leftarrow \infty$

$dist(i, j, k) \leftarrow f(dist(\cdot, \cdot, k-1))$.



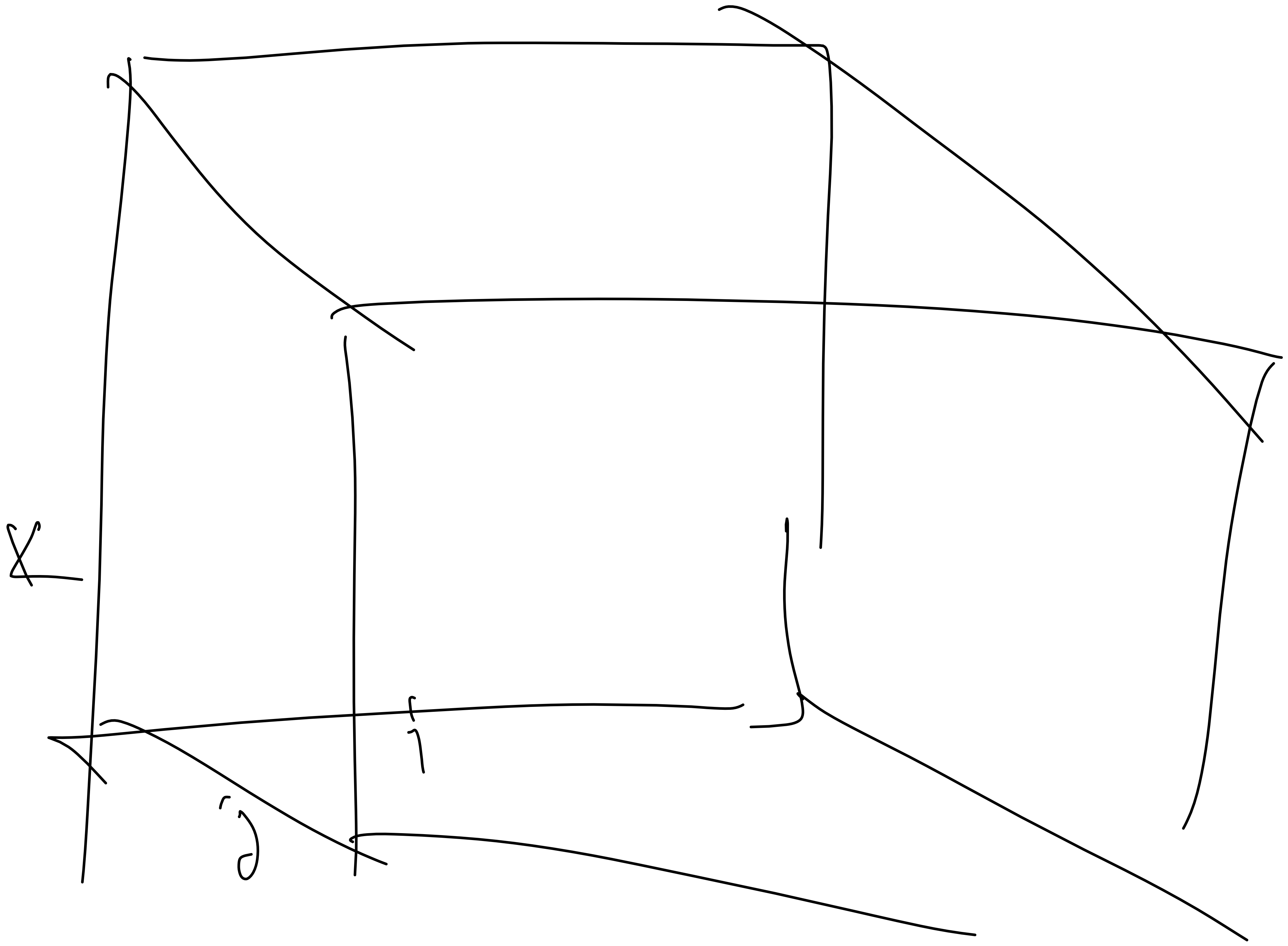
$\{1, 2, \ldots, k\}$

k being used.

$dist(i, k, k-1) + dist(k, j, k-1)$

k not being used.

$dist(i, j, k-1)$


X
1
2

// Initialize here for $k=0$.

for $k = 1$ to n

for $i = 1$ to n

for $j = 1$ to n.

$$\text{dist}(i,j,k) = \min \begin{cases} \text{dist}(i,j,k-1), \\ \text{dist}(i,k,k-1) + \text{dist}(k,j,k-1) \end{cases}$$

$$\text{largest}(i,j,\cancel{k}) = \text{either}$$

$$\text{largest}(i,j,\cancel{k-1})$$

or ~~largest (i,~~

k.