Unit 4- Floyd's Algorithm

All pairs shortest path

All pairs shortest path

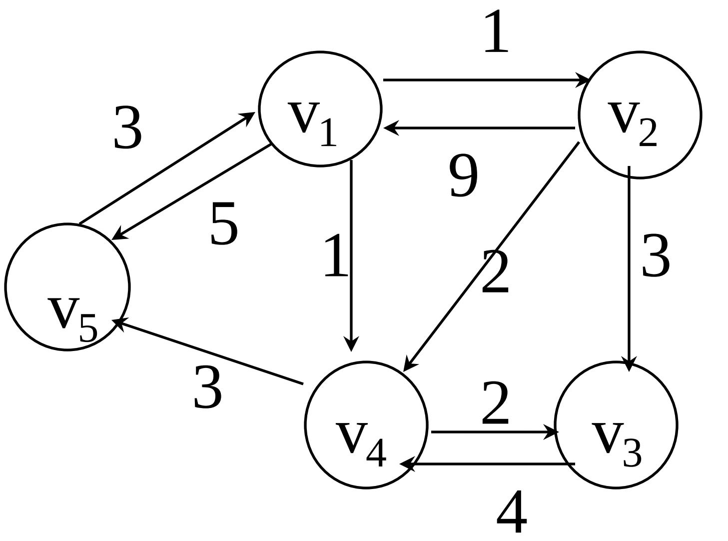
- ***The problem:*** **find the shortest path between every pair of vertices of a graph**

- ***The graph*****: may contain negative edges but no negative cycles**

- ***A representation*****: a weight matrix where**
  W(i,j)=0 if i=j.
  W(i,j)=∞ if there is no edge between i and j.
  W(i,j)="weight of edge"
- **Note: we have shown principle of optimality applies to shortest path problems**

The weight matrix and the graph

	1	2	3	4	5
1	0	1	∞	1	5
2	9	0	3	2	∞
3	∞	∞	0	4	∞
4	∞	∞	2	0	3
5	3	∞	∞	∞	0



The subproblems

- **How can we define the shortest distance $d_{i,j}$ in terms of "smaller" problems?**
- **One way is to restrict the paths to only include vertices from a restricted subset.**
- **Initially, the subset is empty.**
- **Then, it is incrementally increased until it includes all the vertices.**

The subproblems

- **Let $D^{(k)}[i,j]$=weight of a shortest path from v_i to v_j using only vertices from $\{v_1,v_2,\ldots,v_k\}$ as intermediate vertices in the path**
  - $D^{(0)}=W$
  - $D^{(n)}=D$ **which is the goal matrix**
- **How do we compute $D^{(k)}$ from $D^{(k\text{-}1)}$?**

The recursive definition

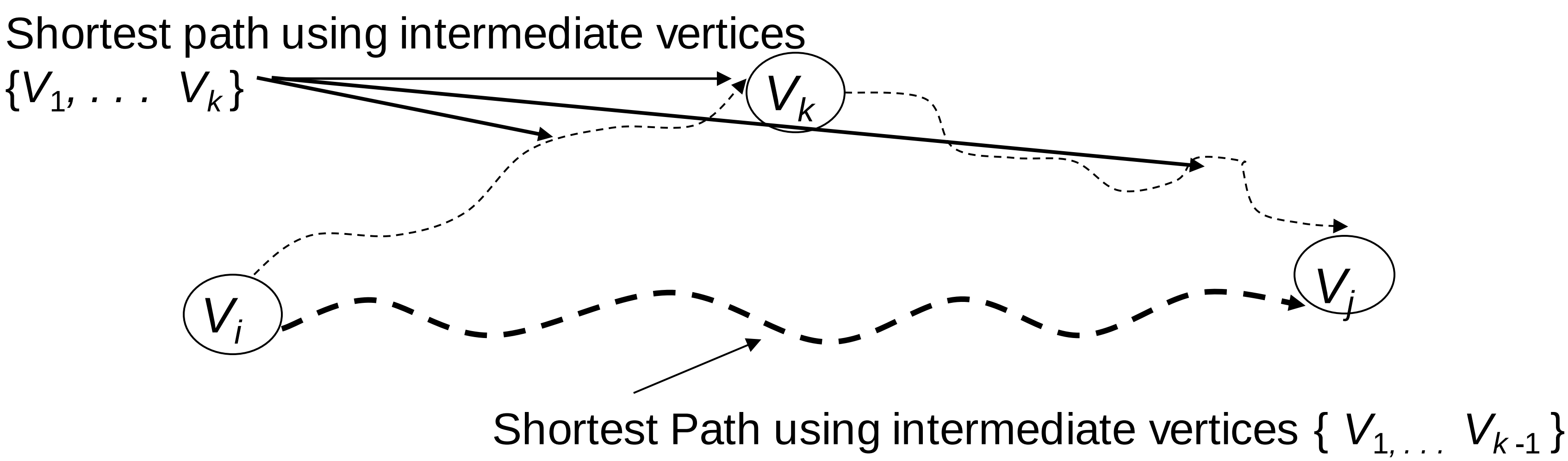
- **Since**

$$D^{(k)}[i,j] = D^{(k-1)}[i,j] \text{ or}$$

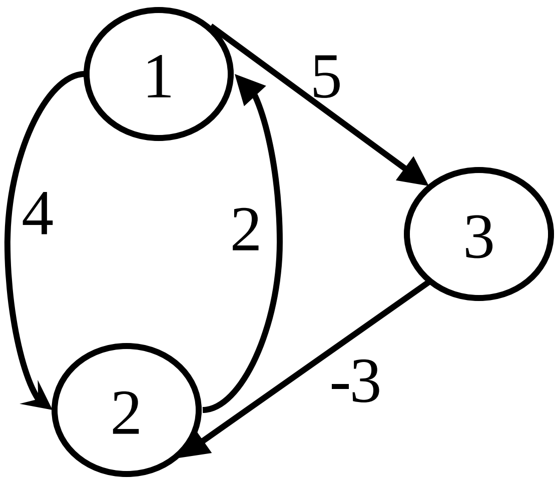
$$D^{(k)}[i,j] = D^{(k-1)}[i,k] + D^{(k-1)}[k,j].$$

We conclude:

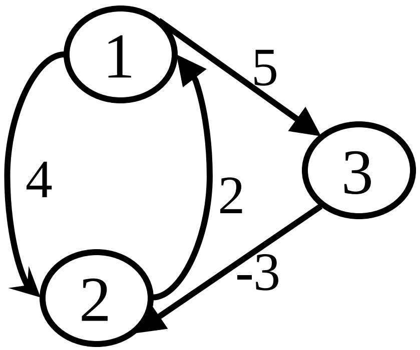
$$D^{(k)}[i,j] = \min\{ D^{(k-1)}[i,j], D^{(k-1)}[i,k] + D^{(k-1)}[k,j] \}.$$



Example



$$W = D^0 = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 4 & 5 \\ 2 & 2 & 0 & \infty \\ 3 & \infty & -3 & 0 \end{array}$$



$D^0 =$

	1	2	3
1	0	4	5
2	2	0	∞
3	∞	-3	0

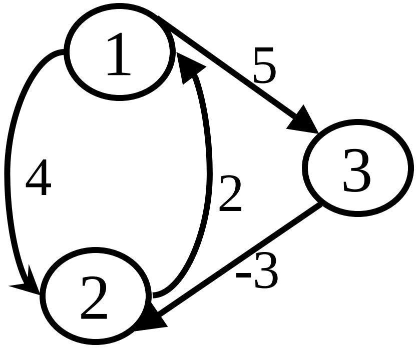
k = 1
Vertex 1 can be intermediate node

$D^1 =$

	1	2	3
1	0	4	5
2	2		
3	∞		

$$D^1[2,3] = \min(D^0[2,3], D^0[2,1]+D^0[1,3])$$
$$= \min (\propto, 7)$$
$$= 7$$

$$D^1[3,2] = \min(D^0[3,2], D^0[3,1]+D^0[1,2])$$
$$= \min (-3,\propto)$$
$$= -3$$



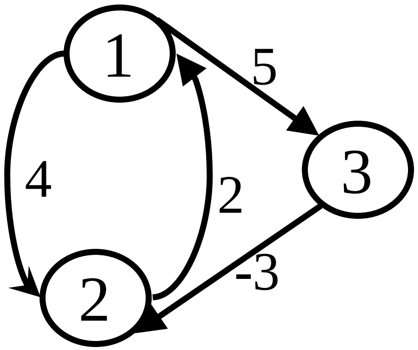
$$D^1 = \begin{array}{c|ccc|} & 1 & 2 & 3 \\ \hline 1 & 0 & 4 & 5 \\ 2 & 2 & 0 & 7 \\ 3 & \infty & -3 & 0 \\ \hline \end{array}$$

k = 2
Vertices 1, 2 can be intermediate

$$D^2 = \begin{array}{c|ccc|} & 1 & 2 & 3 \\ \hline 1 & & 4 & \\ 2 & 2 & 0 & 7 \\ 3 & & -3 & \\ \hline \end{array}$$

$$\begin{aligned} D^2[1,3] &= \min(D^1[1,3], D^1[1,2]+D^1[2,3]) \\ &= \min (5, 4+7) \\ &= 5 \end{aligned}$$

$$\begin{aligned} D^2[3,1] &= \min(D^1[3,1], D^1[3,2]+D^1[2,1]) \\ &= \min (\propto, -3+2) \\ &= -1 \end{aligned}$$



$$D^2 = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 4 & 5 \\ 2 & 2 & 0 & 7 \\ 3 & -1 & -3 & 0 \end{array}$$

k = 3
Vertices 1, 2, 3 can be intermediate

$$D^3 = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & & & 5 \\ 2 & & & 7 \\ 3 & -1 & -3 & 0 \end{array}$$

$$D^3[1,2] = \min(D^2[1,2], D^2[1,3]+D^2[3,2]) = \min(4, 5+(-3)) = 2$$

$$D^3[2,1] = \min(D^2[2,1], D^2[2,3]+D^2[3,1]) = \min(2, 7+(-1)) = 2$$

Floyd's Algorithm: Using 2 D matrices

Floyd

1. ***D*** **←** ***W*** // initialize *D* array to *W* [ ]

2. ***P*** ← 0 // initialize P array to [0]

3. **for *k* ← 1 to *n***

// Computing D' from D

4. do for *i* ← 1 to *n*

5. do for *j* ← 1 to *n*

6. if (*D*[*i*, *j*] > *D*[*i*, *k*] + *D*[*k*, *j*])

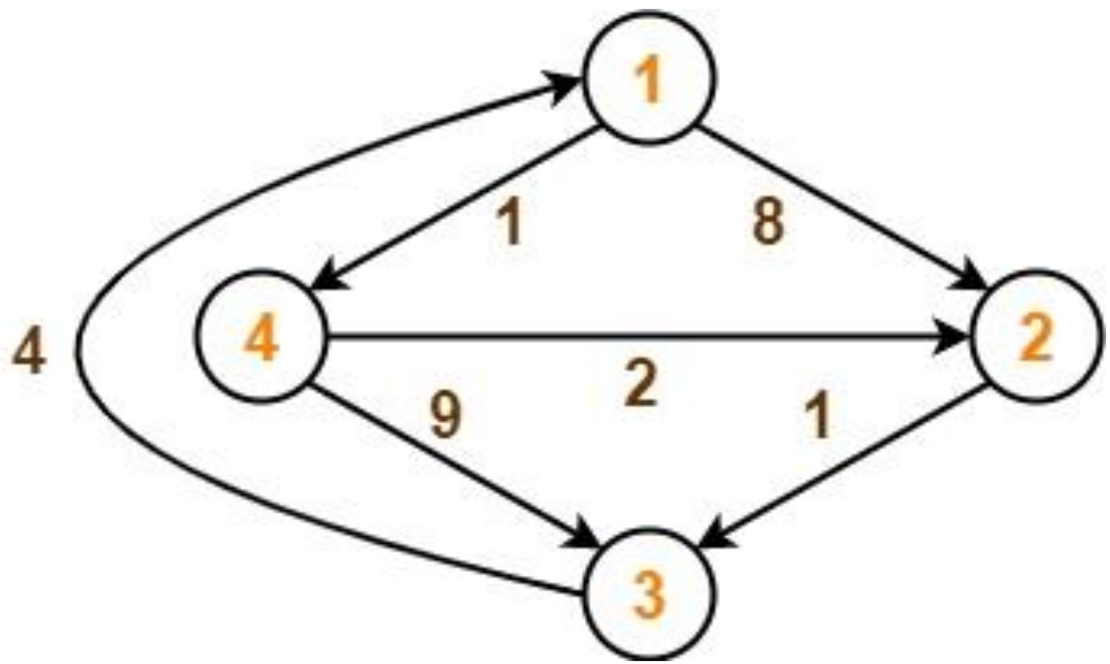
7. then *D'*[*i*, *j*] ← *D*[*i*, *k*] + *D*[*k*, *j*]

8. *P*[*i*, *j*] ← *k*;

9. else *D'*[*i*, *j*] ← *D*[*i*, *j*]

10. *Move D' to D.*

Solve the following example using Floyd Warshal algorithm.



$$D_0 = \begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 8 & \infty & 1 \\ \infty & 0 & 1 & \infty \\ 4 & \infty & 0 & \infty \\ \infty & 2 & 9 & 0 \end{bmatrix} \end{array}$$

$$D_1 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 8 & \infty & 1 \\ \infty & 0 & 1 & \infty \\ 4 & 12 & 0 & 5 \\ \infty & 2 & 9 & 0 \end{bmatrix} \end{array}$$

$$D_2 = \begin{array}{c c} & \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 8 & 9 & 1 \\ \infty & 0 & 1 & \infty \\ 4 & 12 & 0 & 5 \\ \infty & 2 & 3 & 0 \end{bmatrix} \end{array}$$

$$D_3 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 0 & 8 & 9 & 1 \\ 5 & 0 & 1 & 6 \\ 4 & 12 & 0 & 5 \\ 7 & 2 & 3 & 0 \end{bmatrix} \end{array}$$

$$D_4 = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 3 & 4 & 1 \\ 2 & 5 & 0 & 1 & 6 \\ 3 & 4 & 7 & 0 & 5 \\ 4 & 7 & 2 & 3 & 0 \end{array}$$