Caminho Mínimo (3)

Zenilton Patrocínio

Caminho Mínimo entre Todos os Vértices

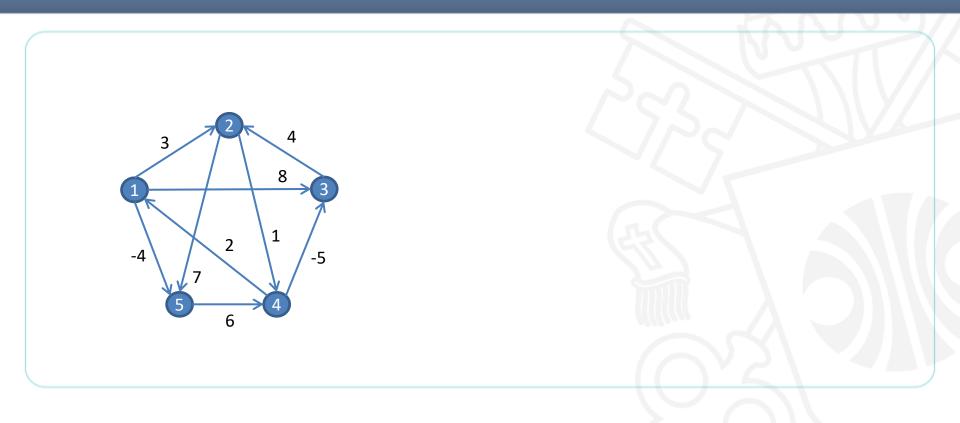
O problema consiste em determinar os caminhos mínimos entre todos os pares de vértices do grafo.

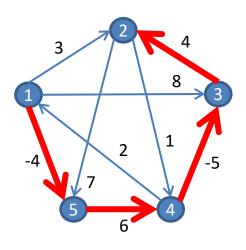
Se todos os <u>custos forem positivos</u>, para n vértices pode-se utilizar n repetições do método de Dijkstra \rightarrow cada vez utiliza-se um vértice como raiz

• Custo $\rightarrow n \times O(n^2) = O(n^3)$ para uma implementação trivial

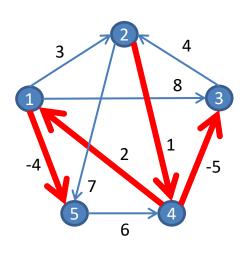
Se <u>houver custos negativos</u> (sem a presença de ciclos negativos), para n vértices pode-se utilizar n repetições do método de Bellman-Ford (com m arestas)

• Custo $\rightarrow n \times O(nm) = O(n^2m) \rightarrow O(n^4) !!!$

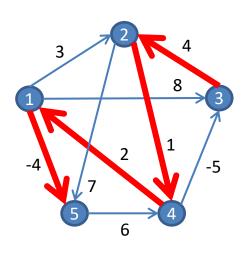




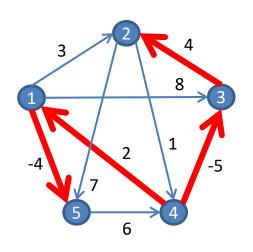
	1	2	3	4	5
1	0	1	-3	2	-4



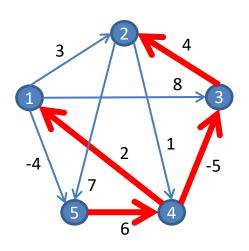
	1	2	3	4	5
1	0	1	-3	2	-4
2	3	0	-4	1	-1



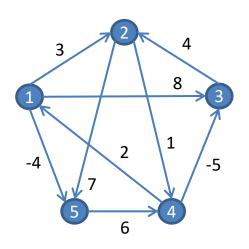
	1	2	3	4	5
1	0	1	-3	2	-4
2	3	0	-4	1	-1
3	7	4	0	5	3



	1	2	3	4	5
1	0	1	-3	2	-4
2	3	0	-4	1	-1
3	7	4	0	5	3
4	2	-1	-5	0	-2



	1	2	3	4	5
1	0	1	-3	2	-4
2	3	0	-4	1	-1
3	7	4	0	5	3
4	2	-1	-5	0	-2
5	8	5	1	6	0



	1	2	3	4	5
1	0	1	-3	2	-4
2	3	0	-4	1	-1
3	7	4	0	5	3
4	2	-1	-5	0	-2
5	8	5	1	6	0

O algoritmo de Floyd-Warshall calcula os caminhos mais curtos entre todos os pares de vértices (consegue lidar arestas de peso negativo desde que não exista ciclo de custo negativo).

Trata-se novamente de um algoritmo de programação dinâmica.

Utiliza o conceito de relaxação do comprimento dos caminhos mais curtos de forma incremental aprimorando a estimativa de distância, até que o valor ótimo seja alcançado.

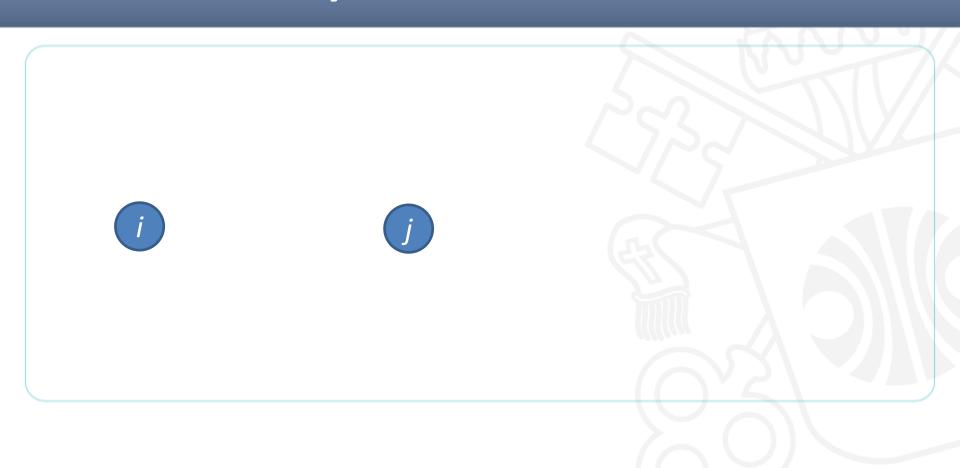
O algoritmo analisa os caminhos entre os vértices i e j passando por cada um dos k vértices intermediários possíveis, $k \in V(G)$.

Considere que os vértices do grafo são numerados de 1 a n.

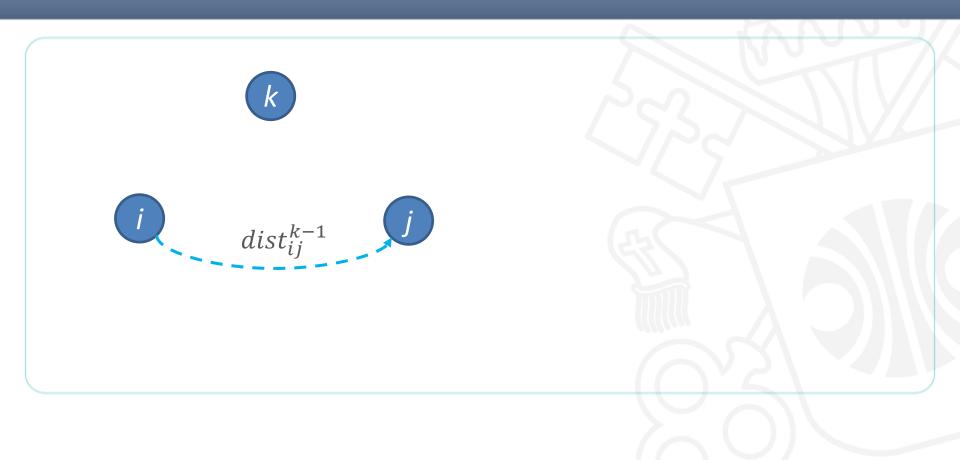
Seja $dist_{ij}^k$ a distância entre o par de vértices i e j usando como intermediários apenas os vértices do conjunto $\{1, 2, ..., k\}$.

Em particular, $dist_{ij}^0 = d_{ij}$, se $\exists (i,j) \in E(G), dist_{ij}^0 = \infty$, caso contrário. Além disso, $dist_{ii}^0 = 0$.

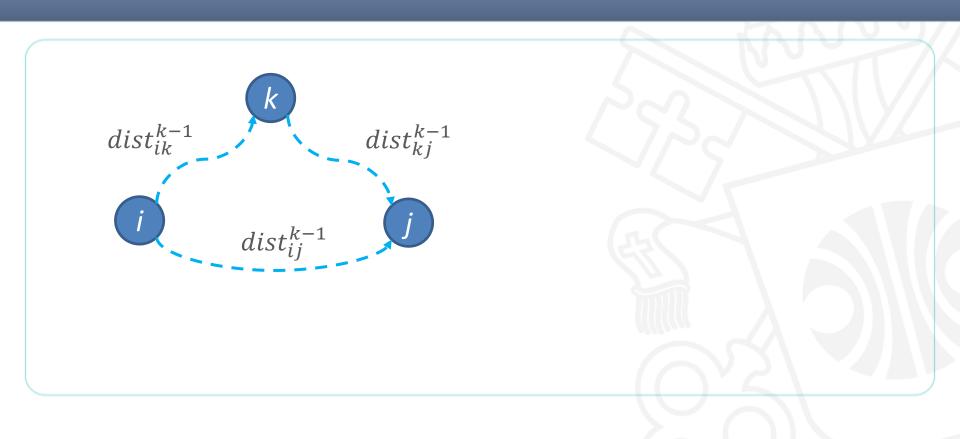
A relaxação do comprimento dos caminhos determina $dist_{ij}^k$ a partir das distâncias $dist_{**}^{k-1}$, isto é, utilizando um caminho de i para j que não passa por k ou o par de caminhos de i para k e de k para j (ambos não passando por k).

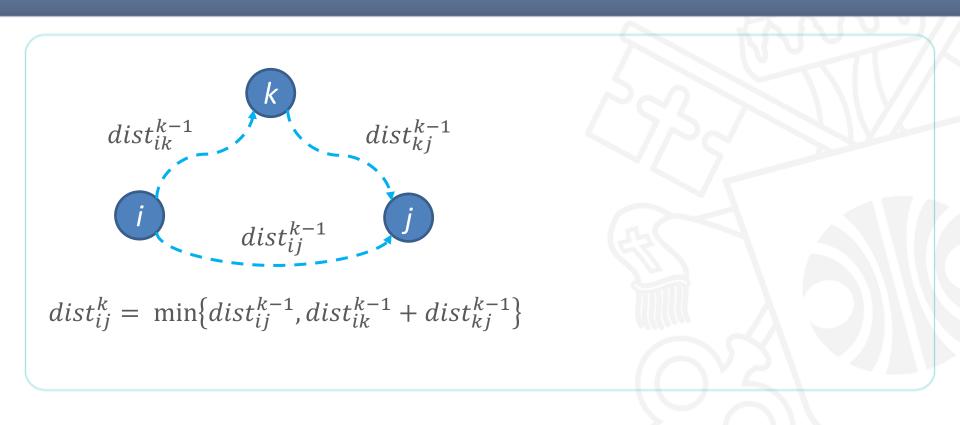


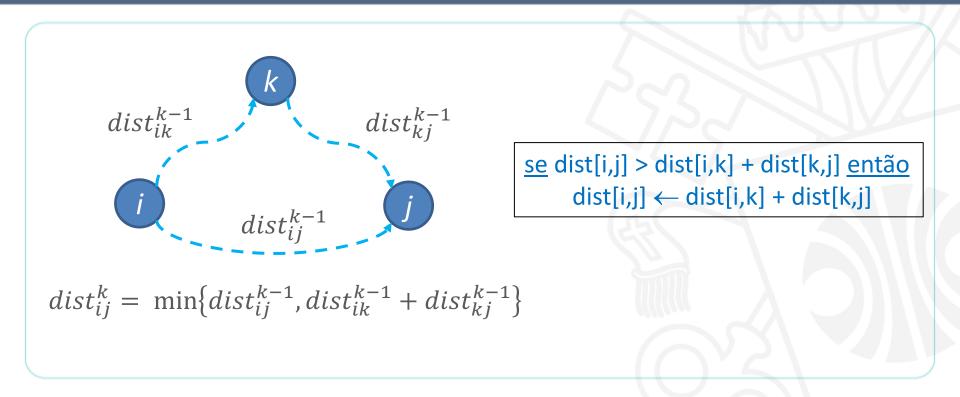


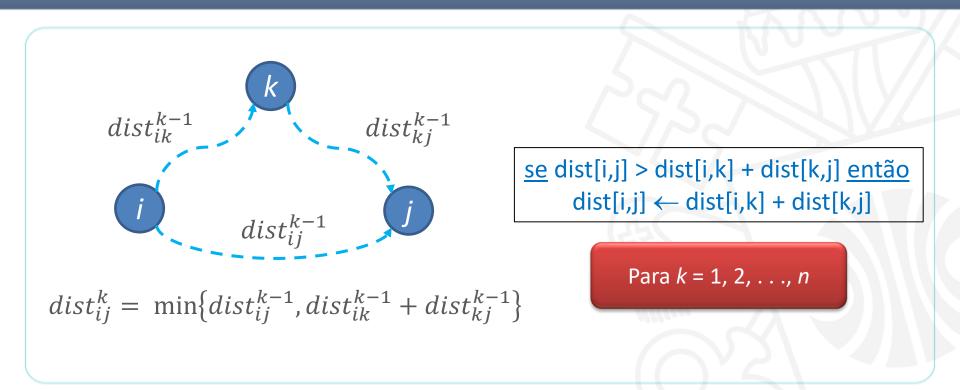






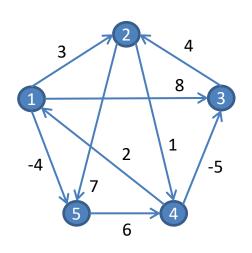






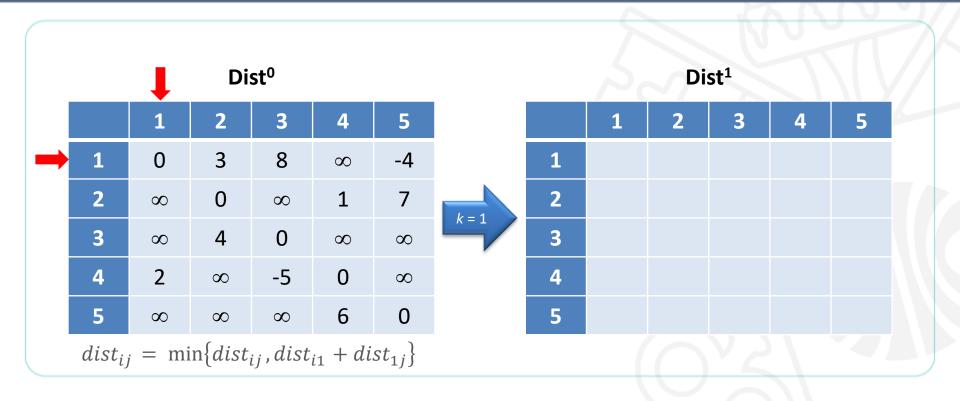
Método de Floyd-Warshall – Algoritmo

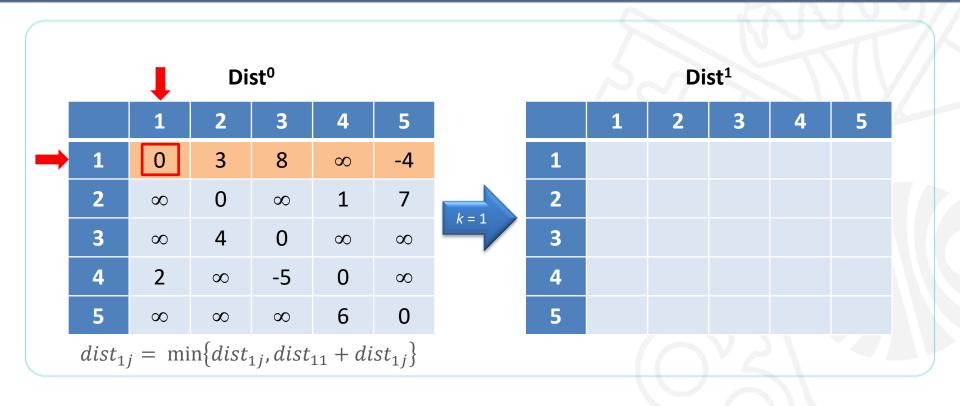
```
1. para i = 1, ..., n faça
                                                                               // Inicializar distâncias
      para j = 1, ..., n \mid j \neq i faça dist[i, j] \leftarrow \infty;
      dist[i, i] \leftarrow 0:
para toda aresta (i, j) \in E(G) faça dist[i, j] \leftarrow d_{ii};
para k = 1, ..., n faça
                                                                 // Para cada possível intermediário
      para i = 1, \ldots, n faça
            para j = 1, ..., n faça
                  se dist[i, j] > dist[i, k] + dist[k, j] então
                                                                       // Testar caminho de i para j
                        dist[i, j] \leftarrow dist[i, k] + dist[k, j];
                                                                              // Atualizar a distância
```

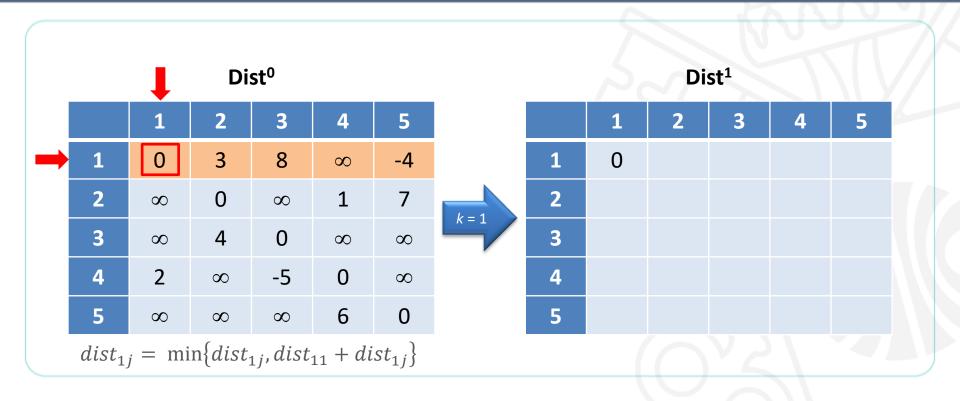


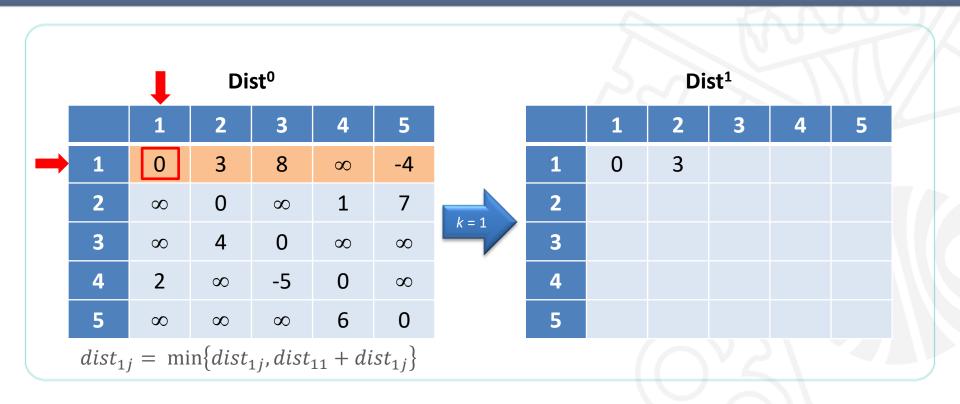
Dist⁰

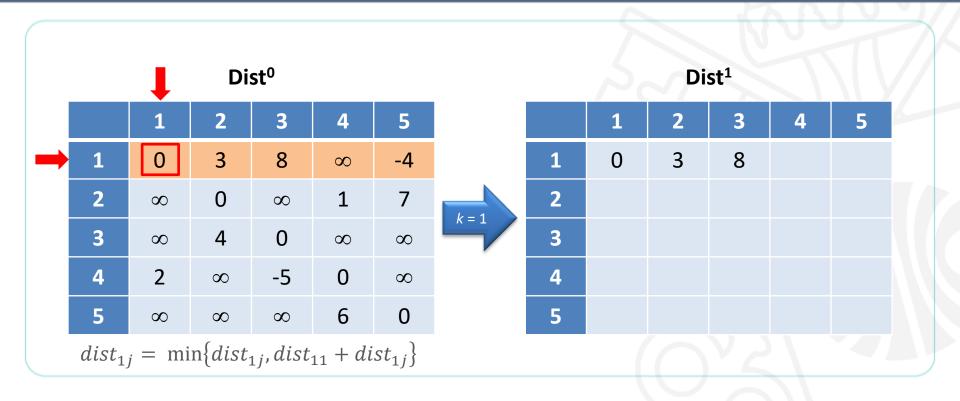
	1	2	3	4	5
1	0	3	8	∞	-4
2	∞	0	∞	1	7
3	∞	4	0	∞	∞
4	2	∞	-5	0	∞
5	∞	∞	∞	6	0

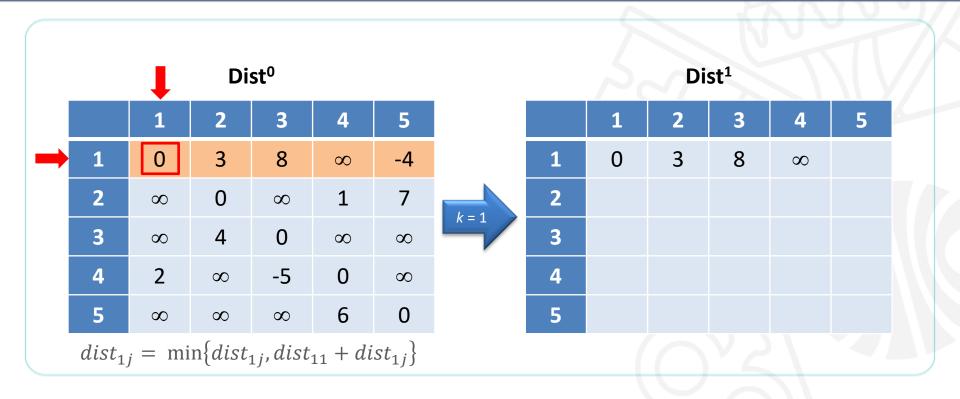


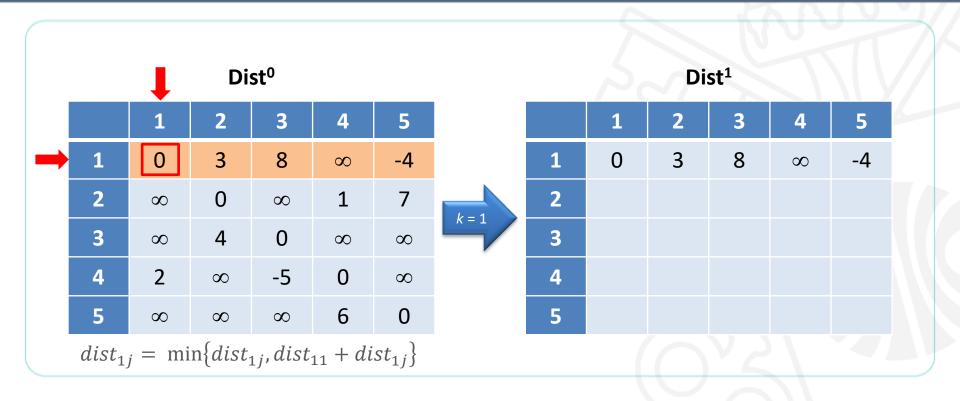


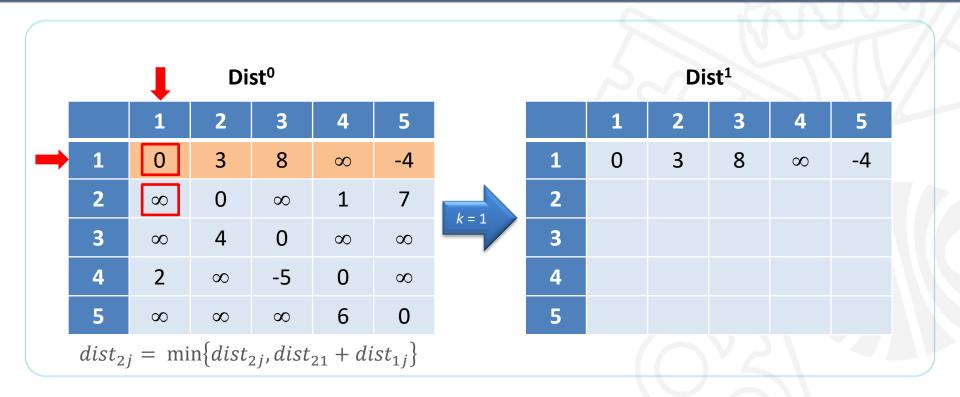


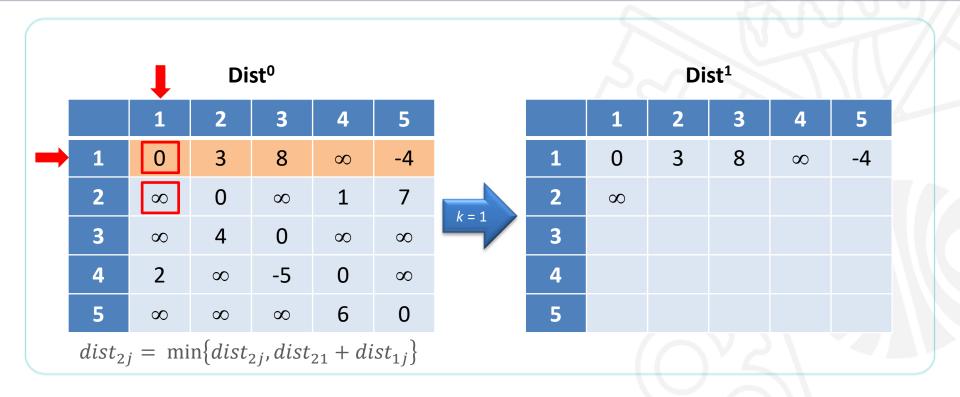


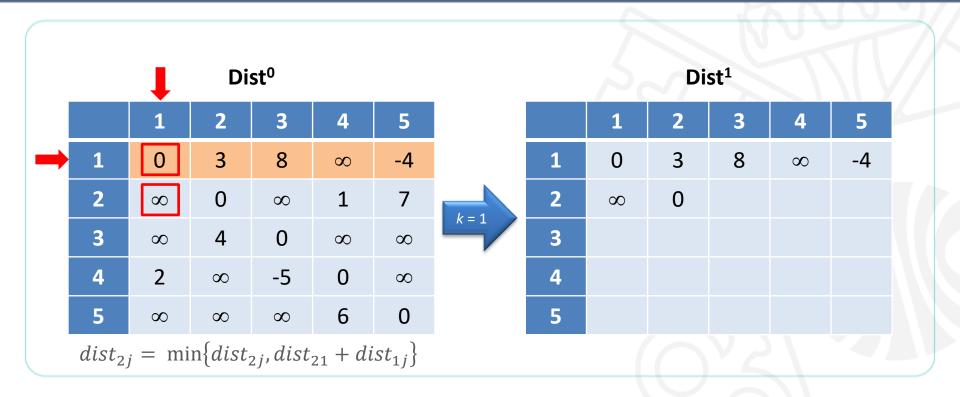


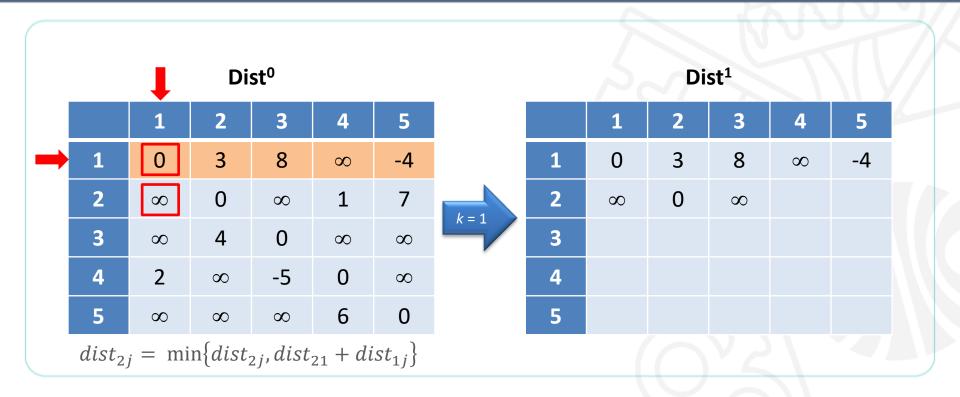


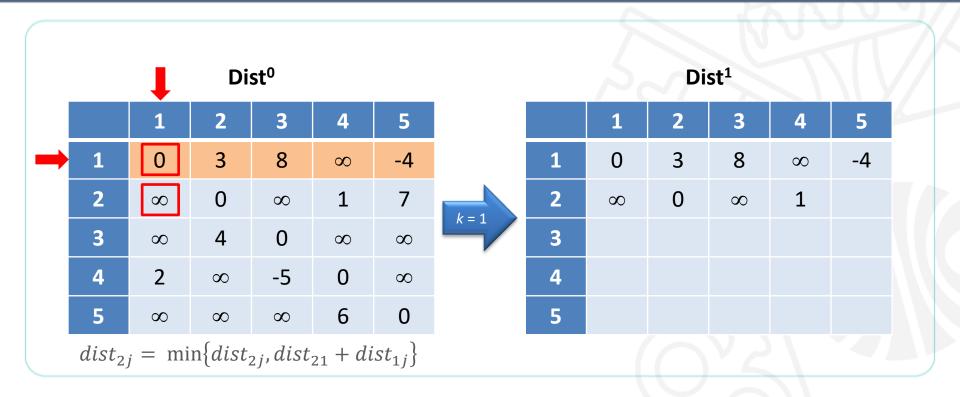


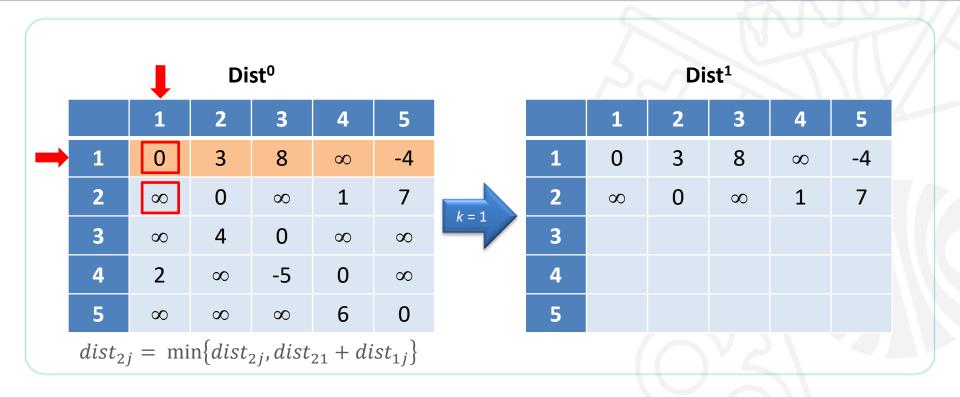


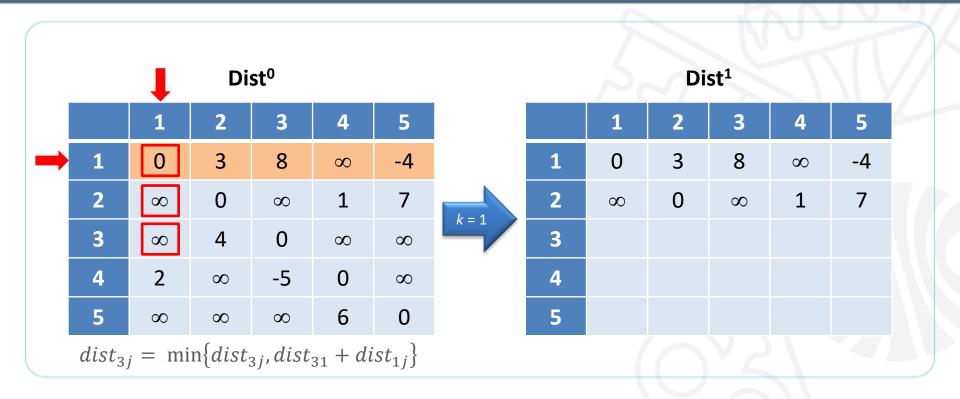


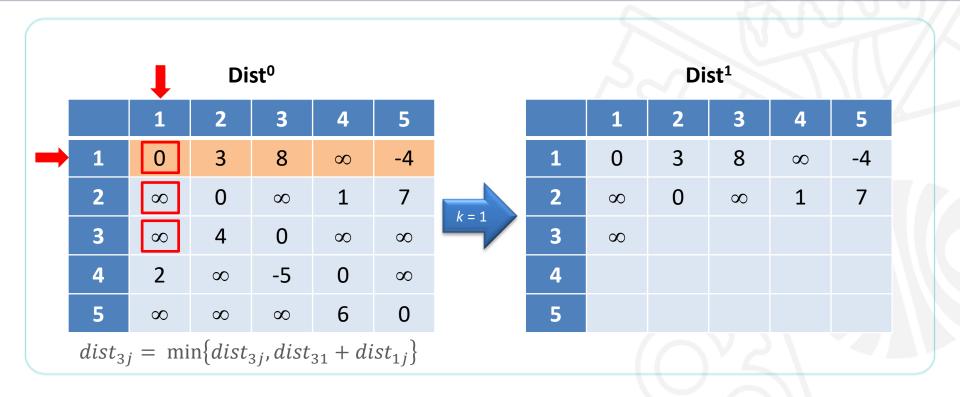


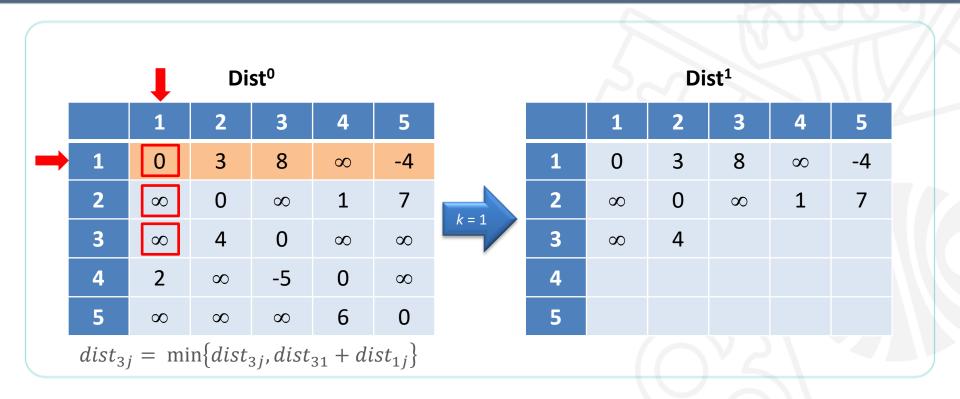


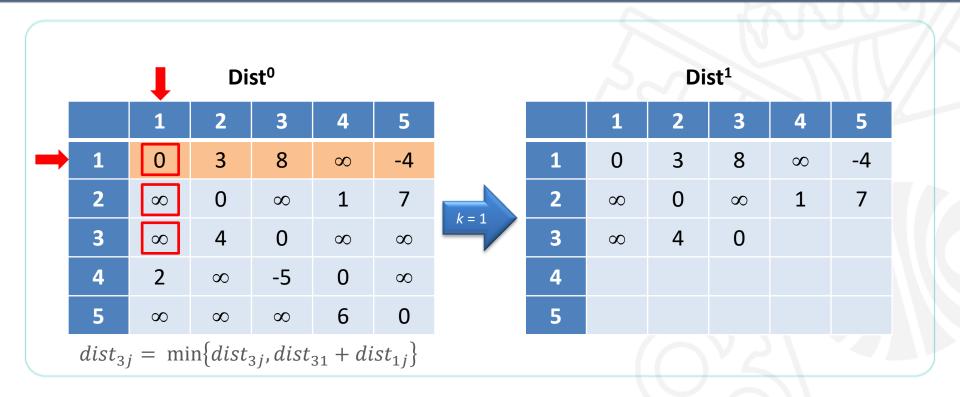


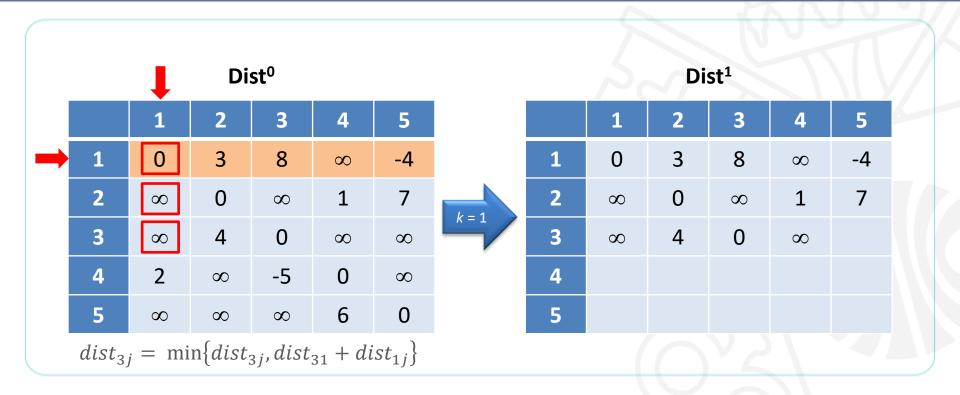


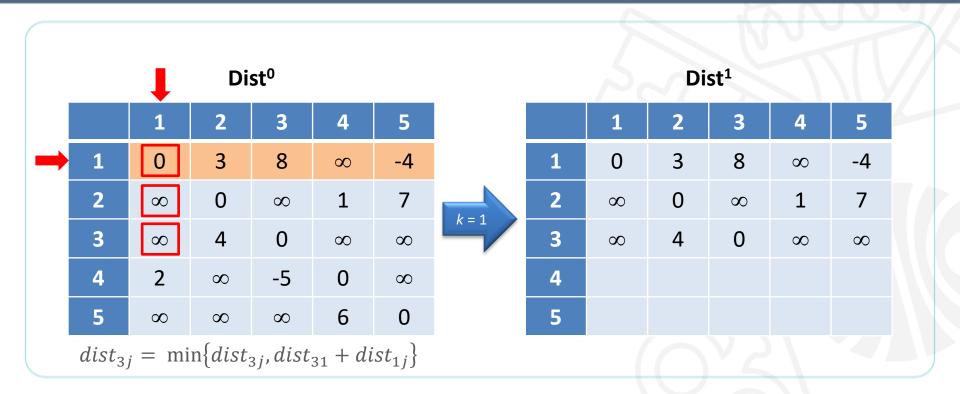


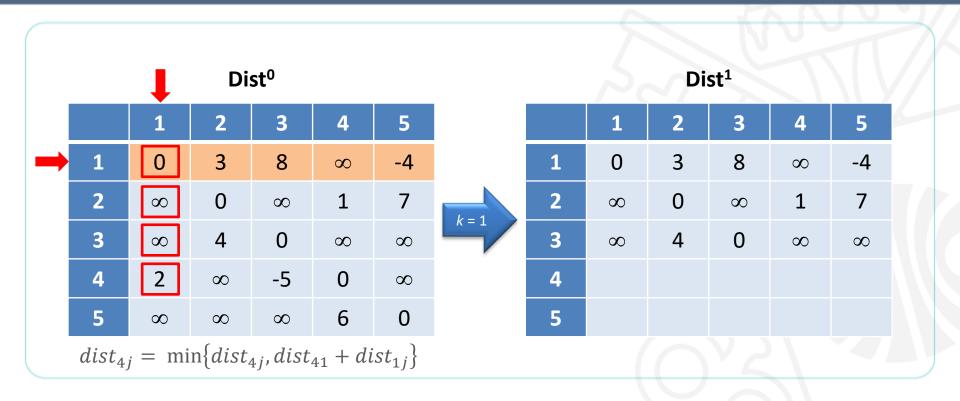


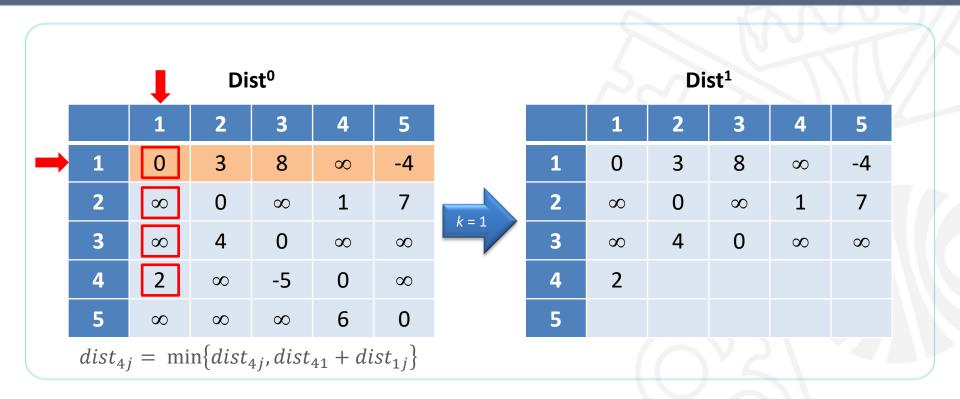


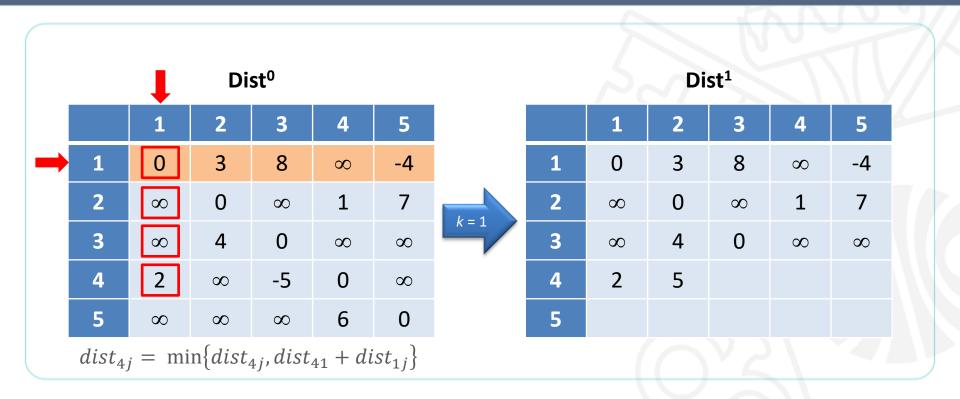


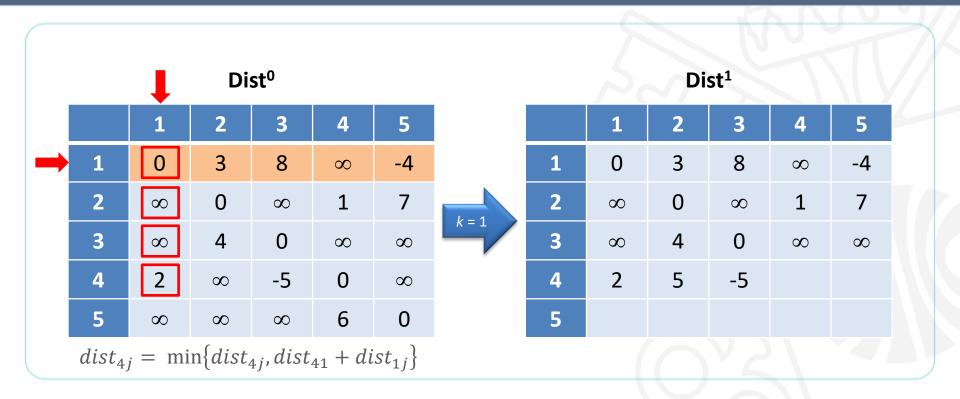


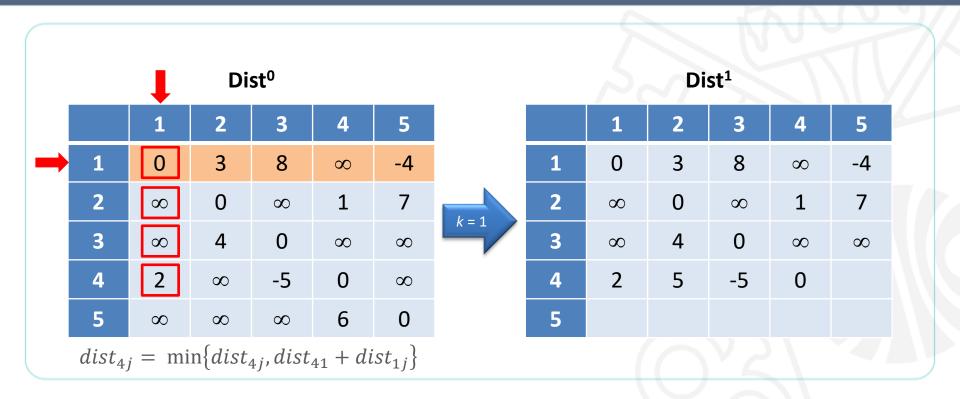


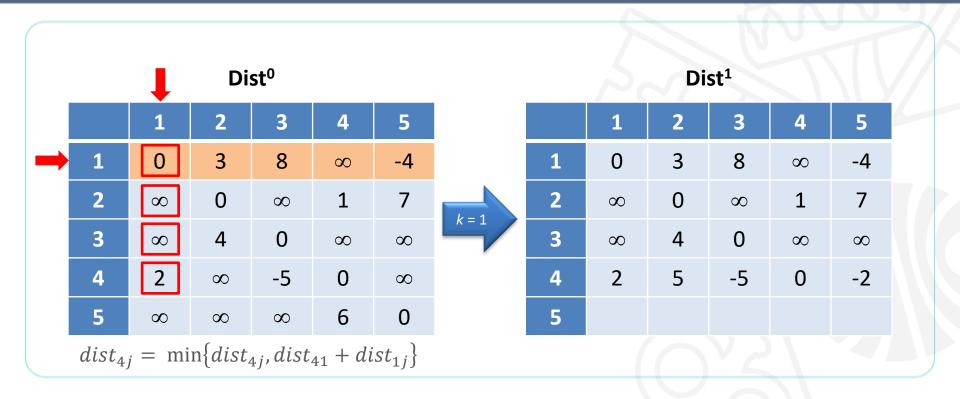


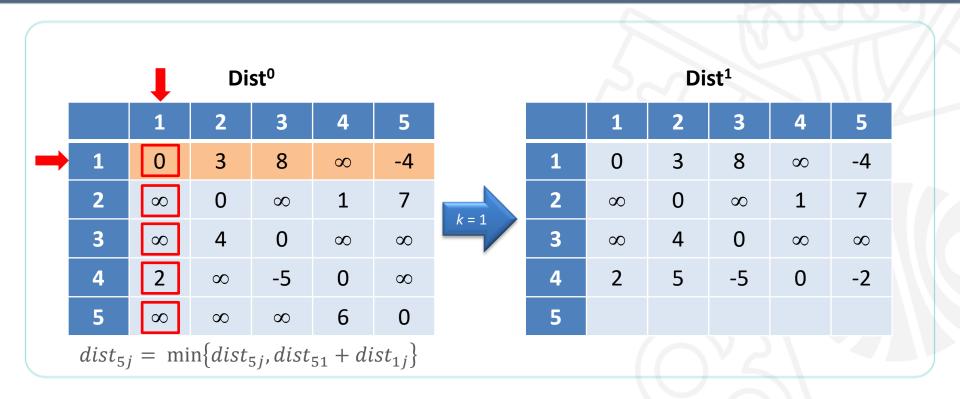


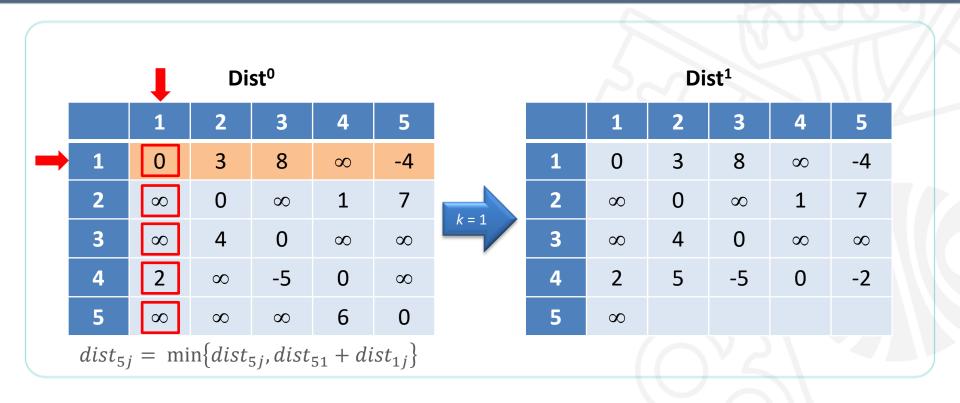


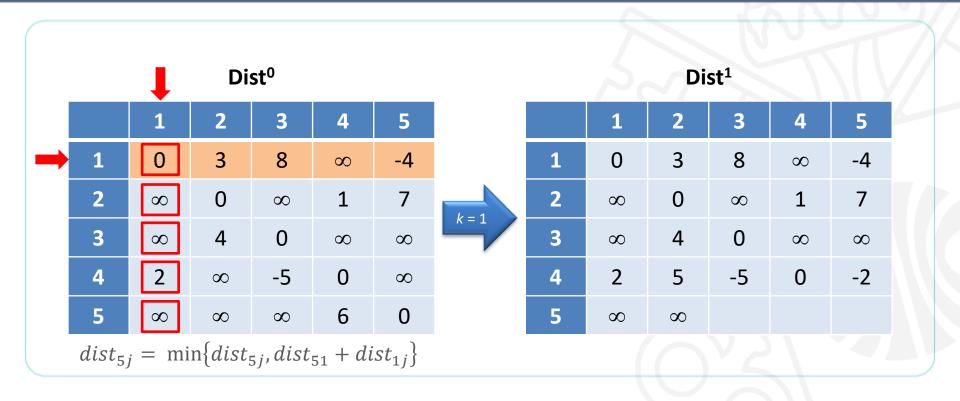


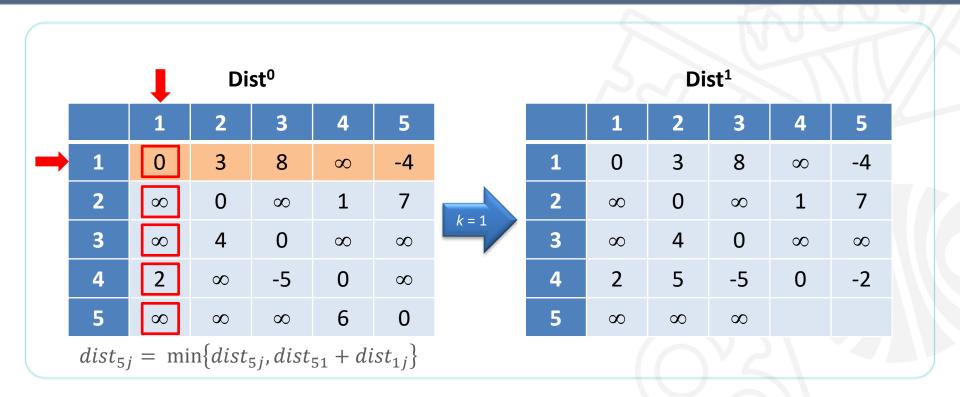


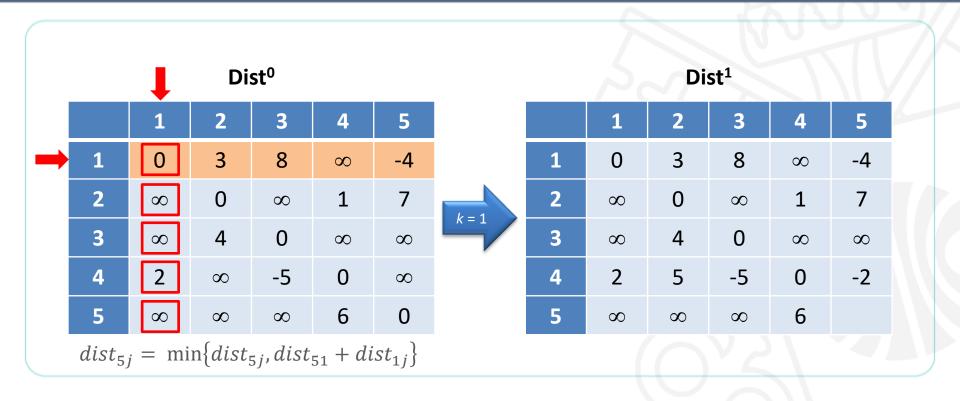


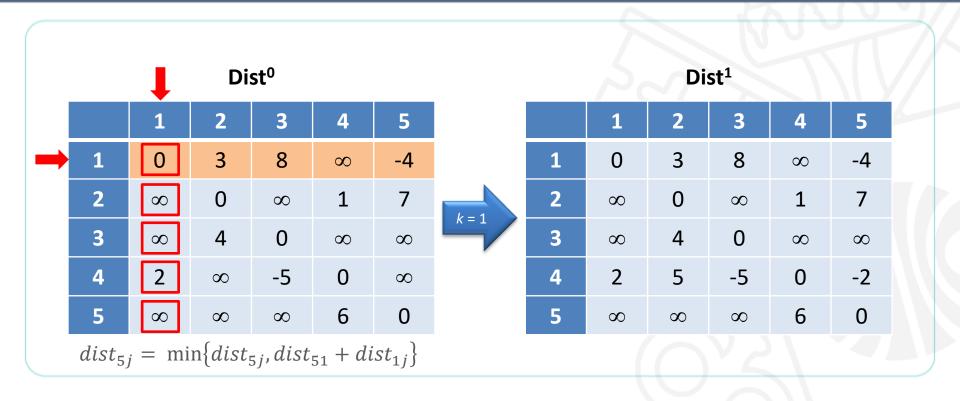


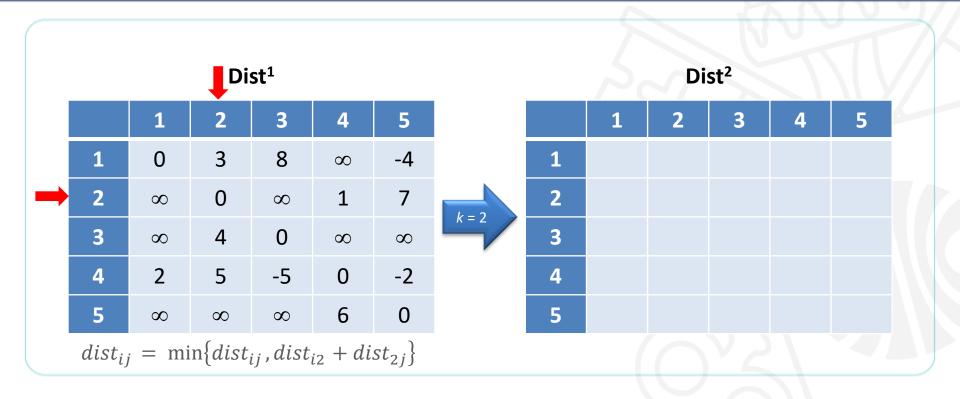


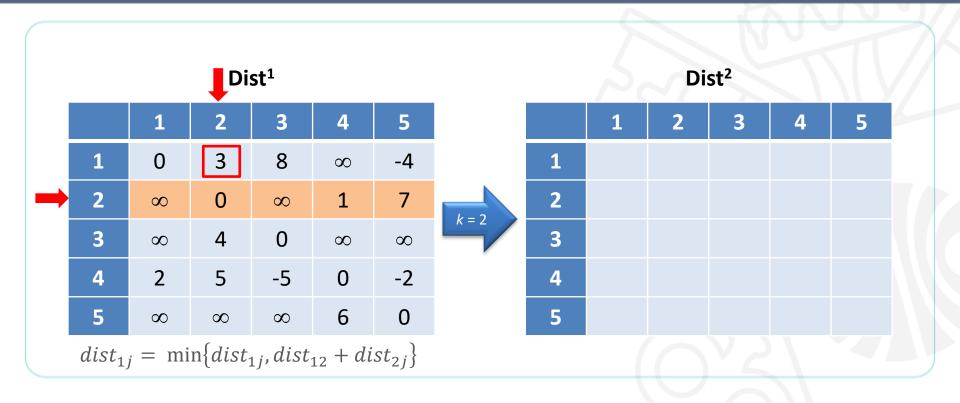


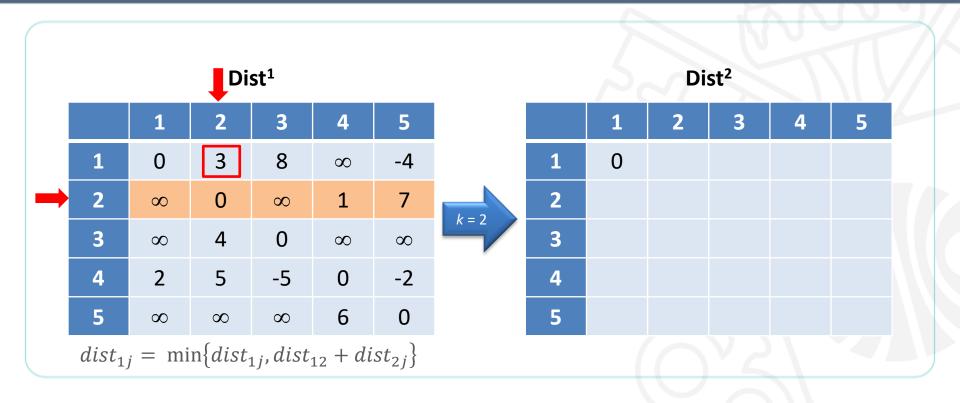


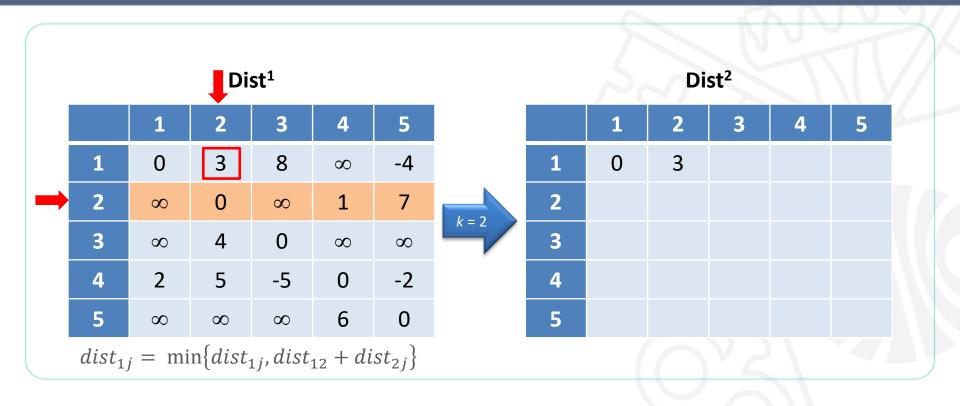


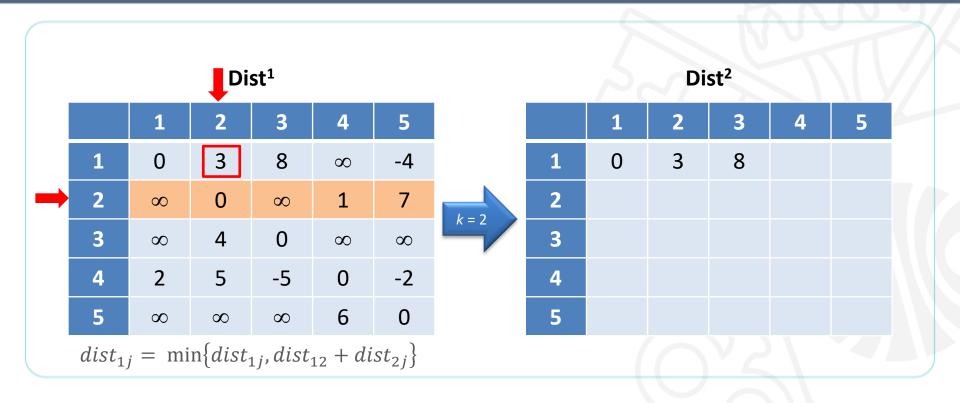


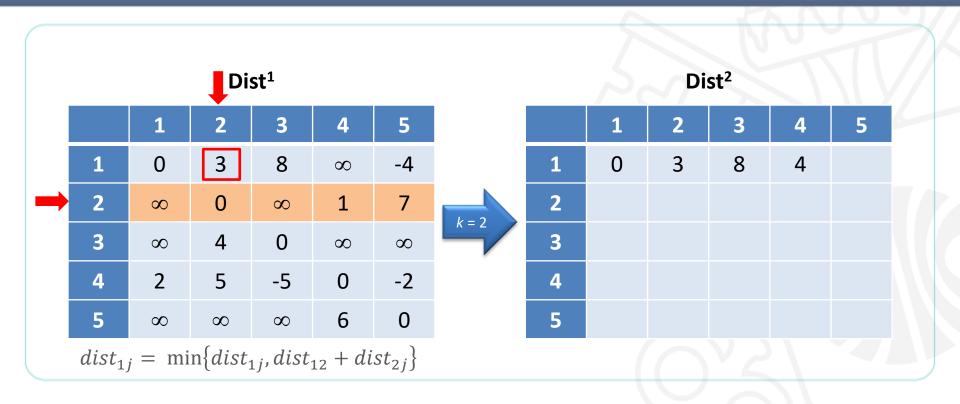


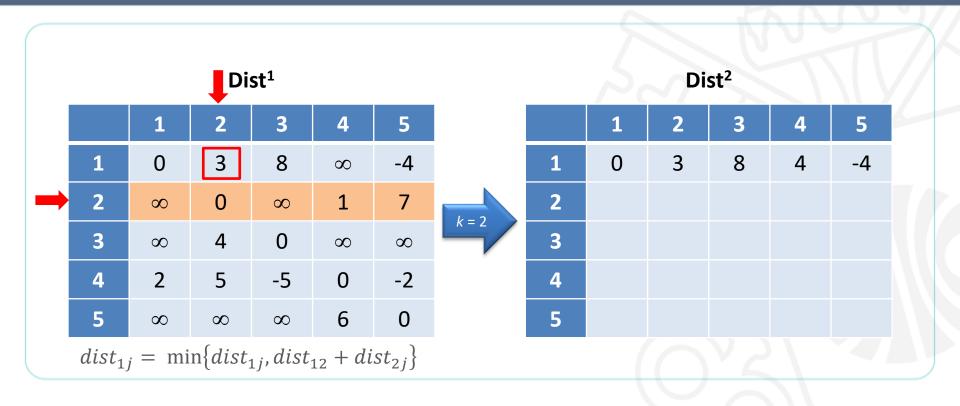


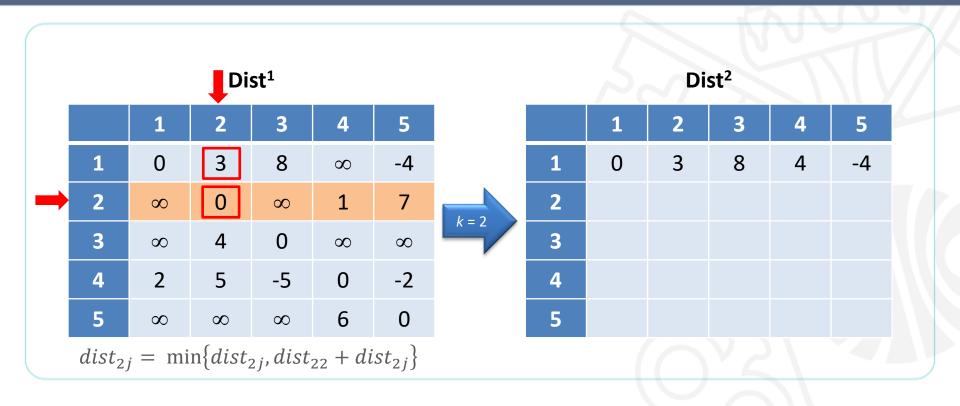


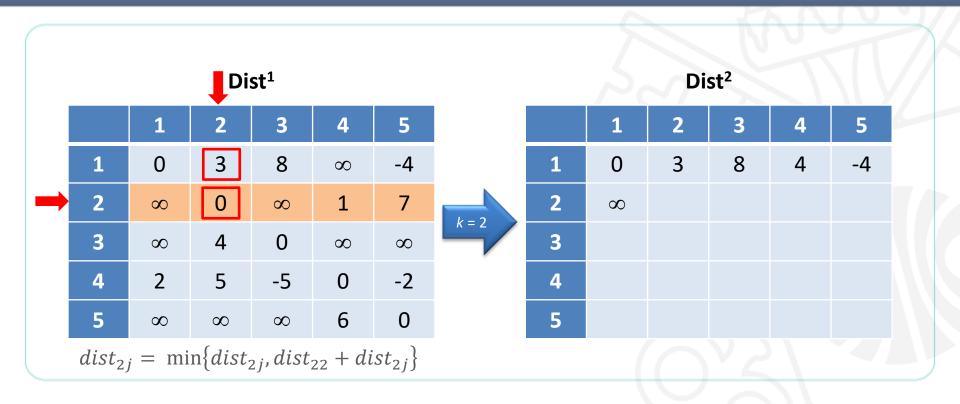


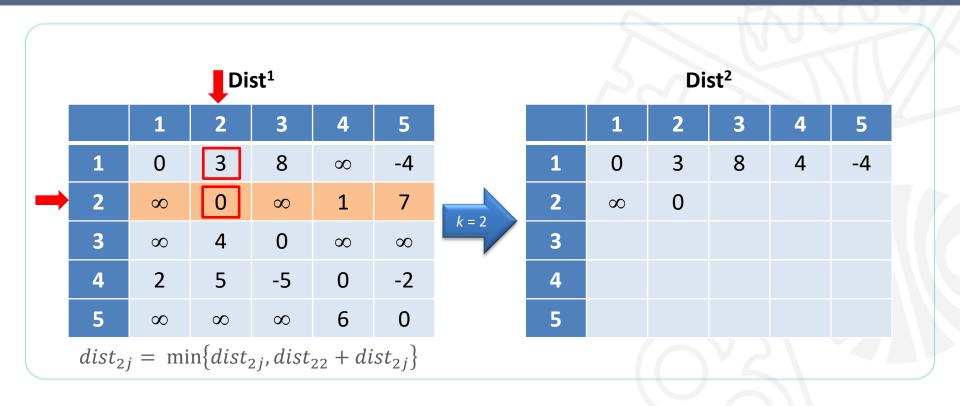


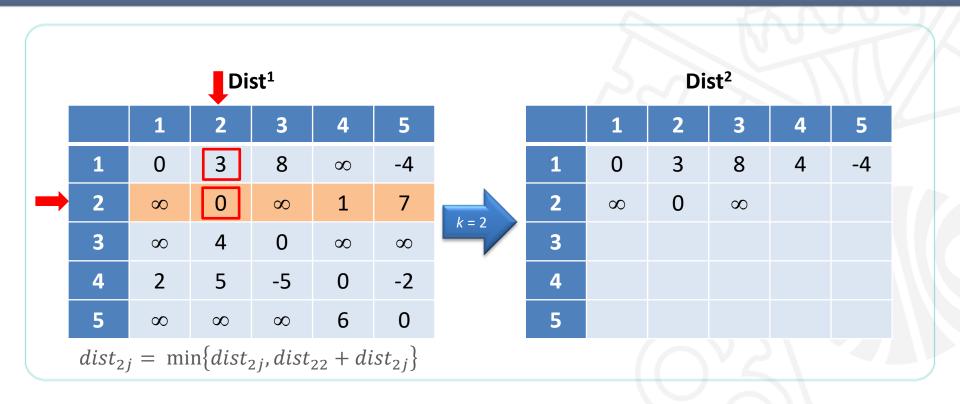


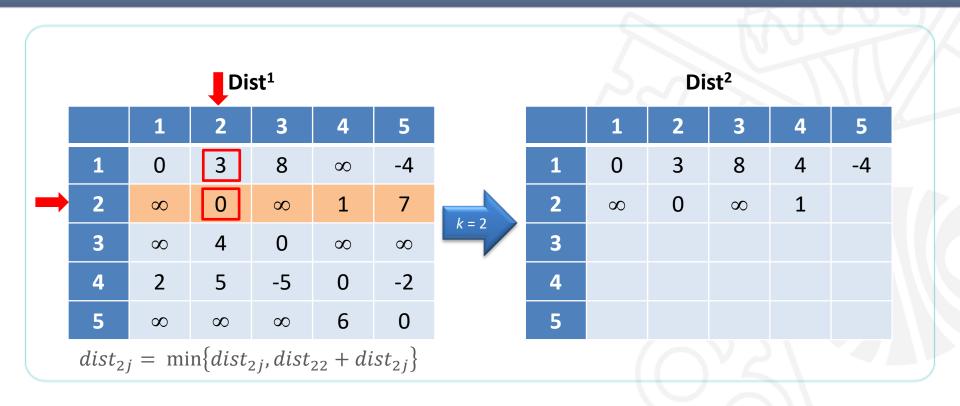


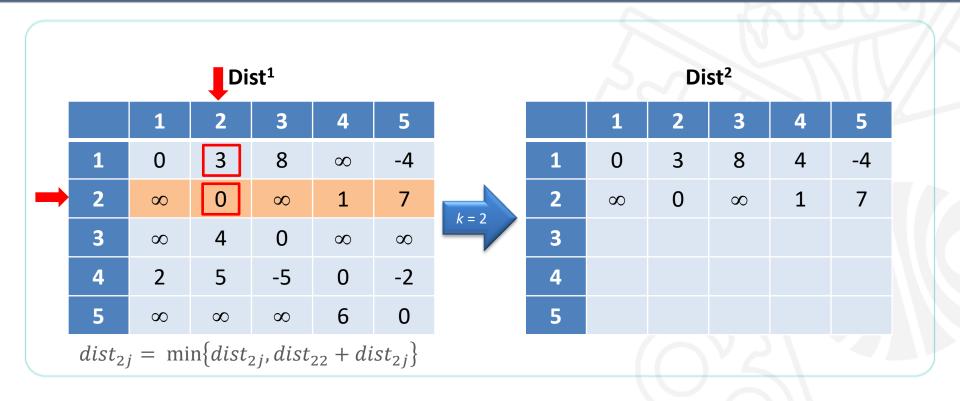


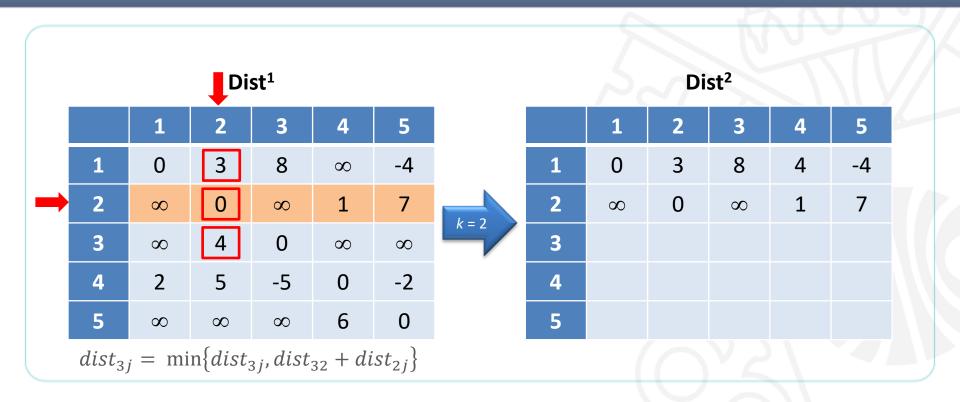


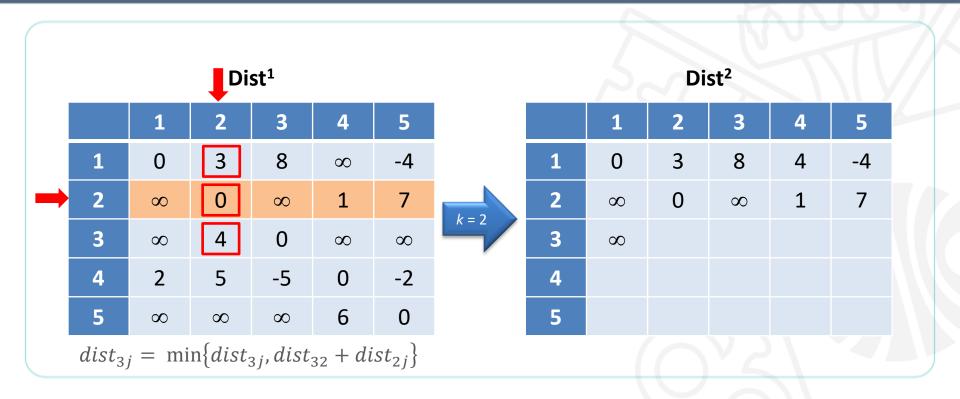


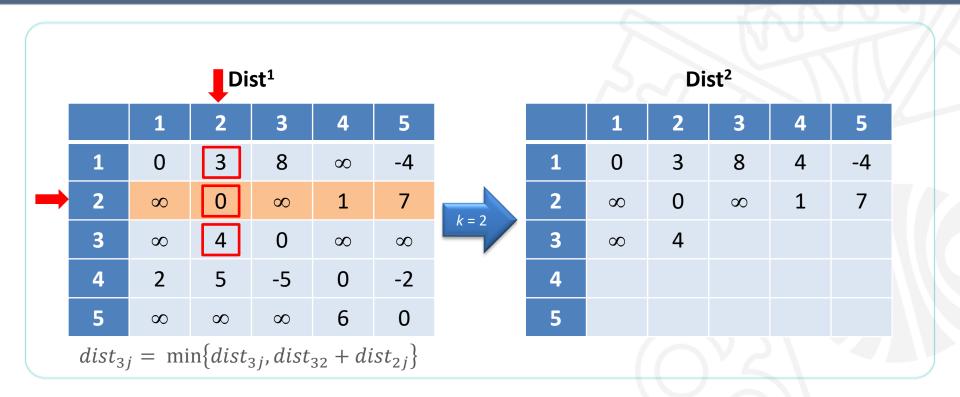


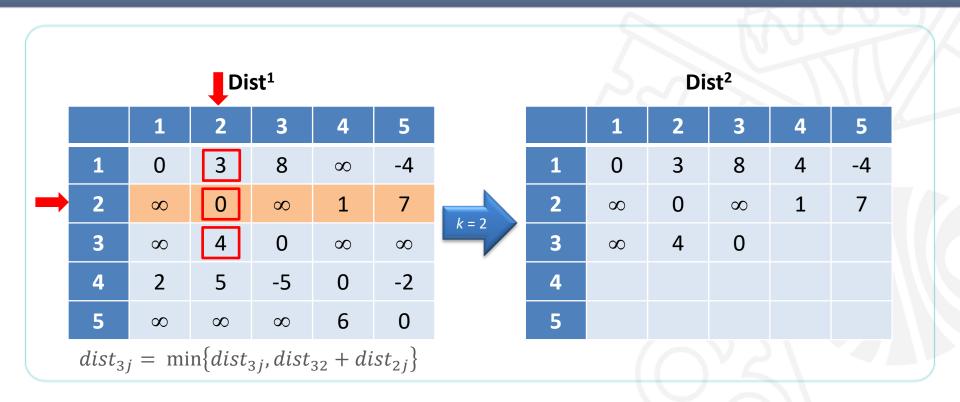


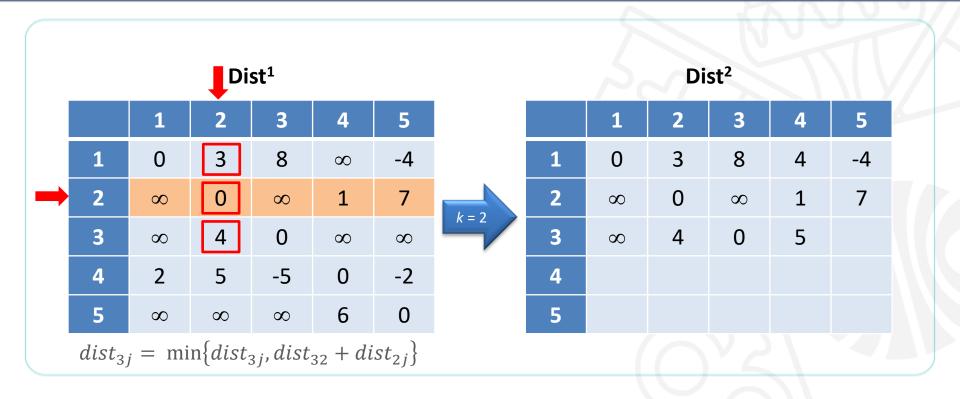


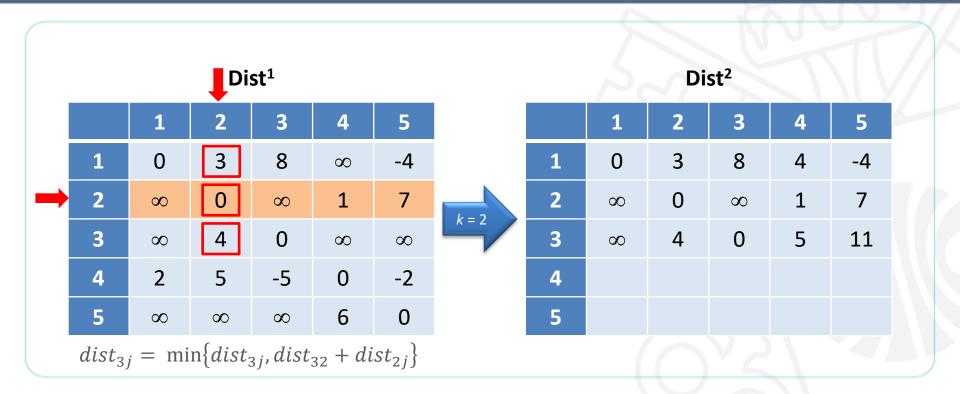


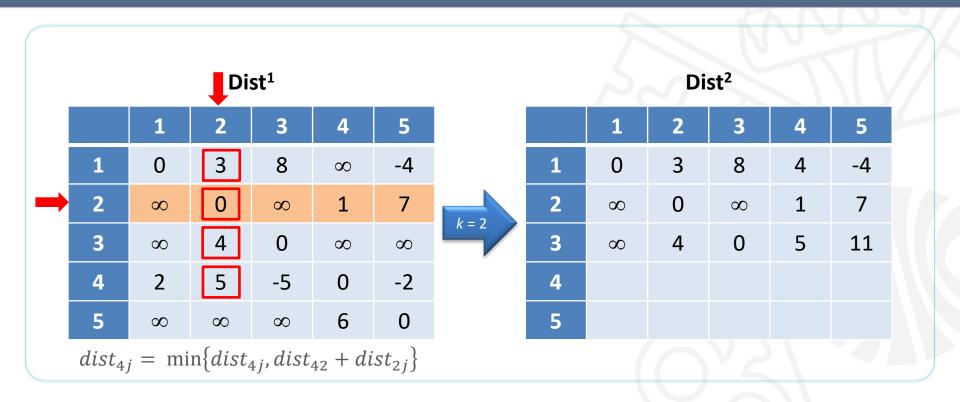


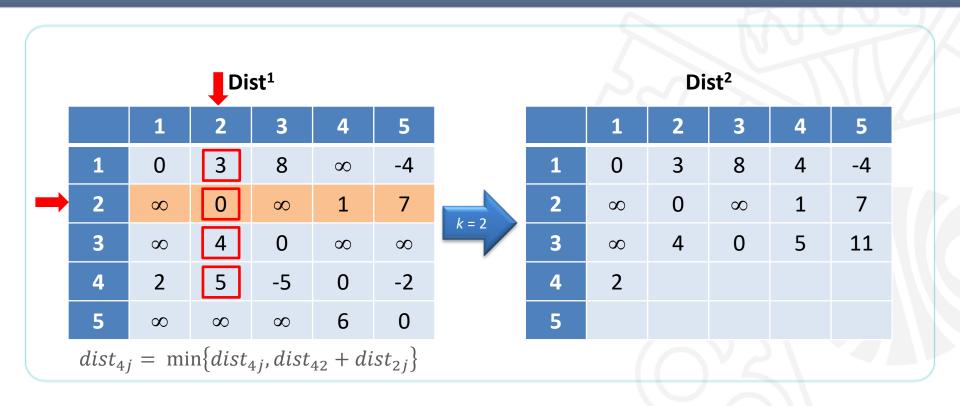


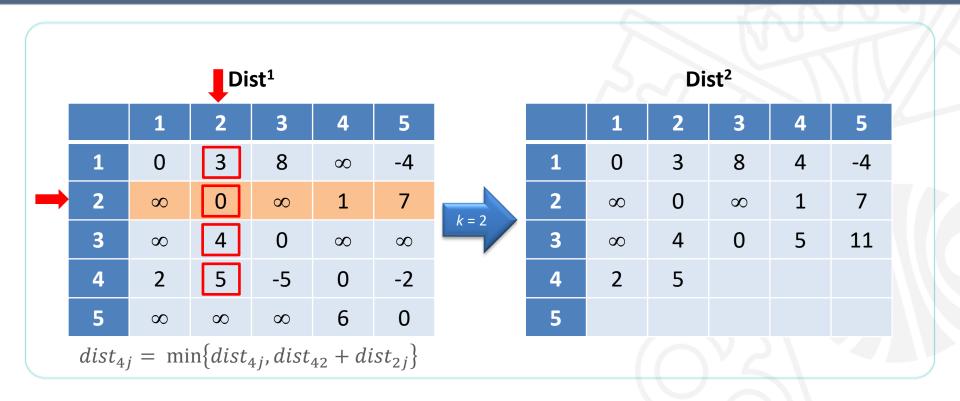


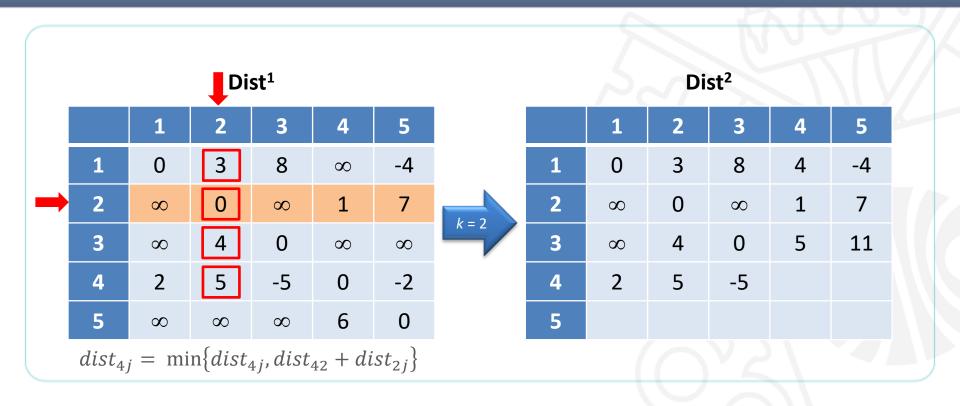


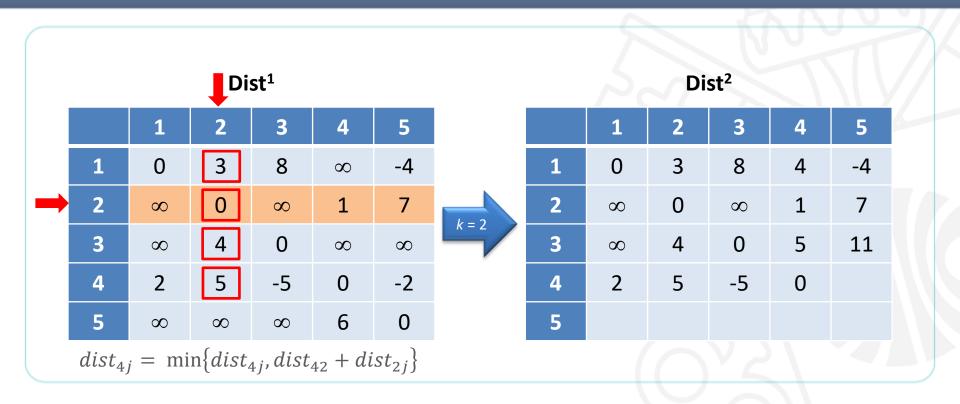


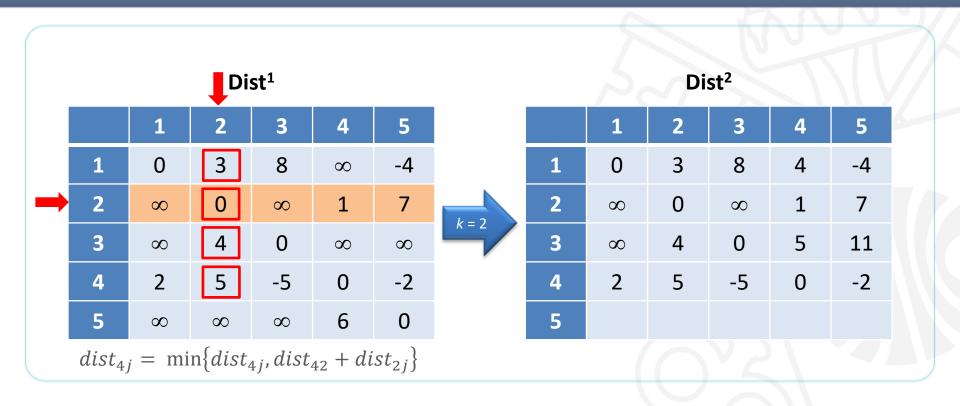


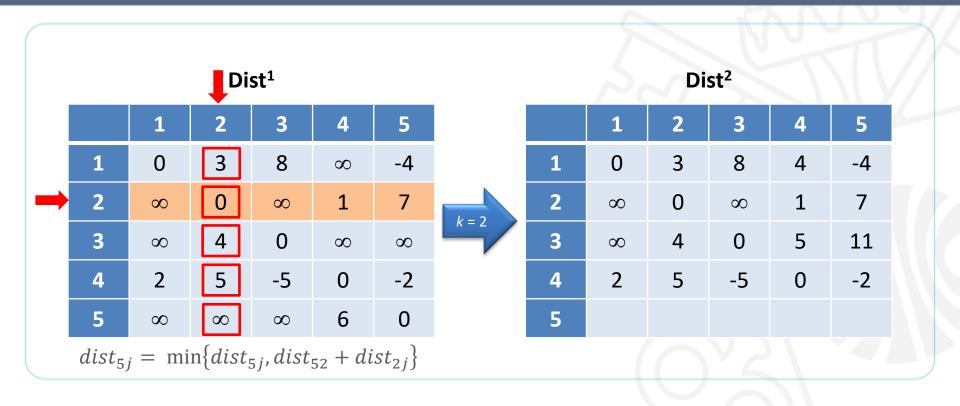


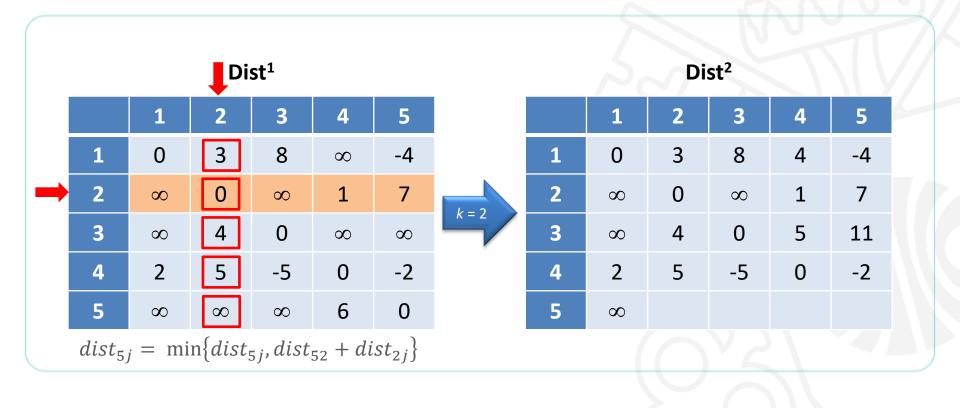


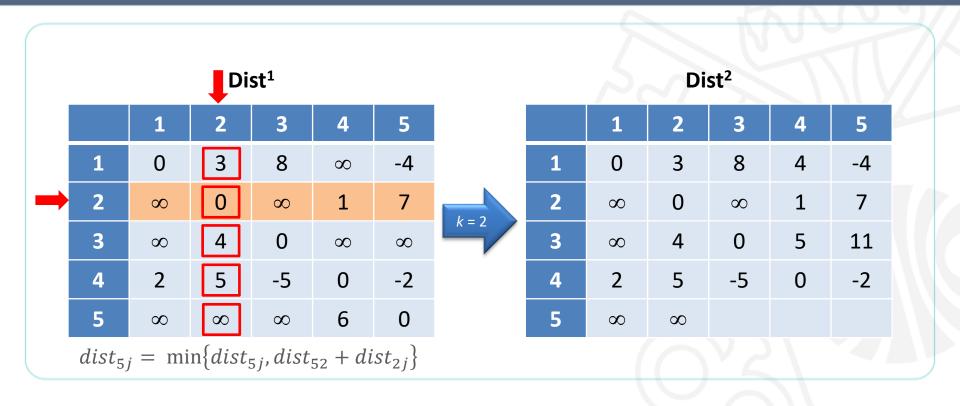


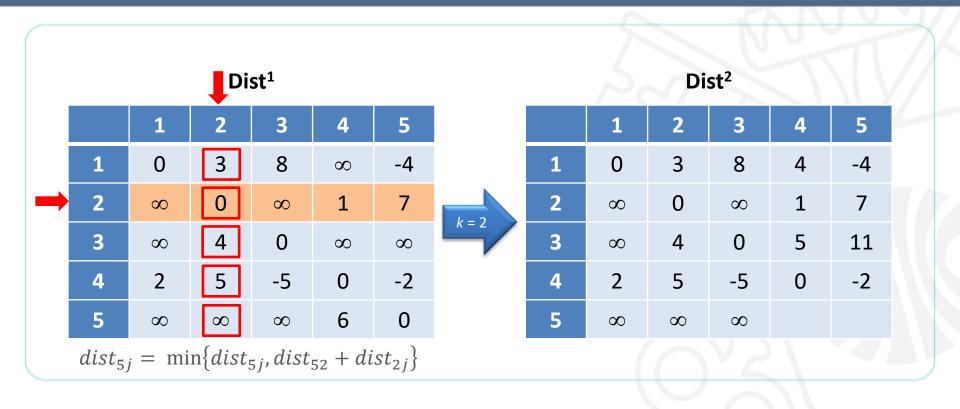


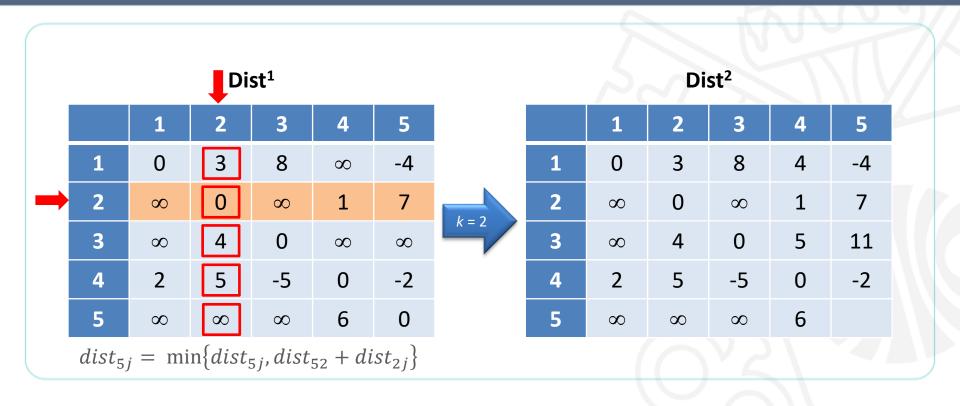


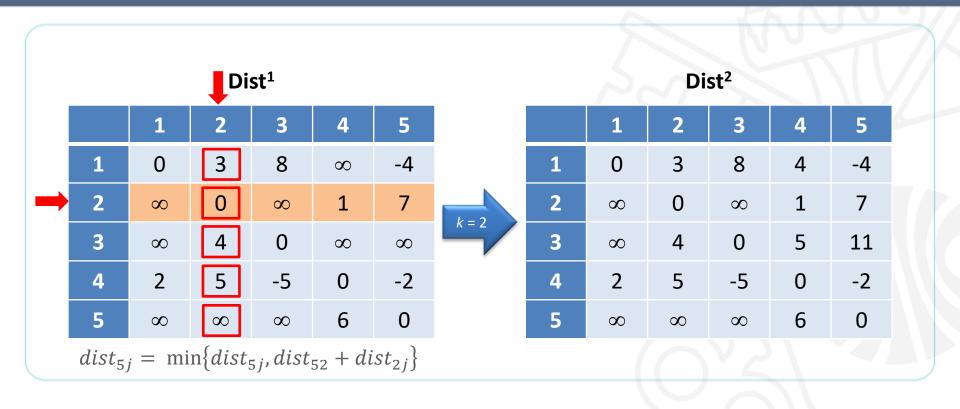


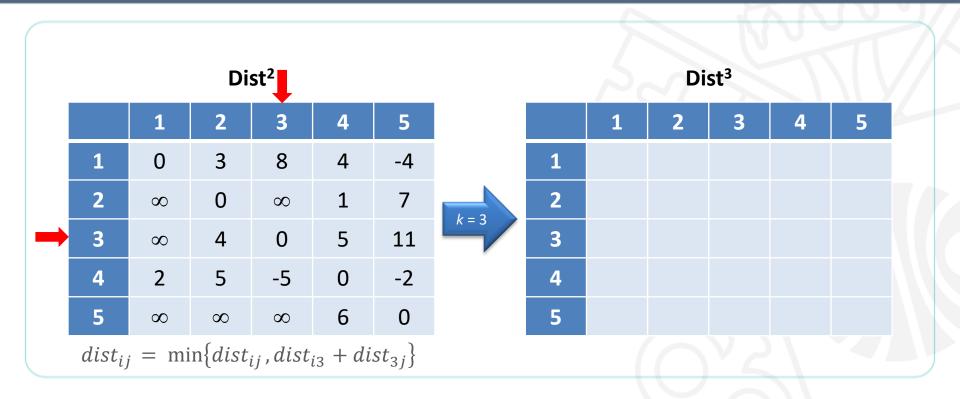


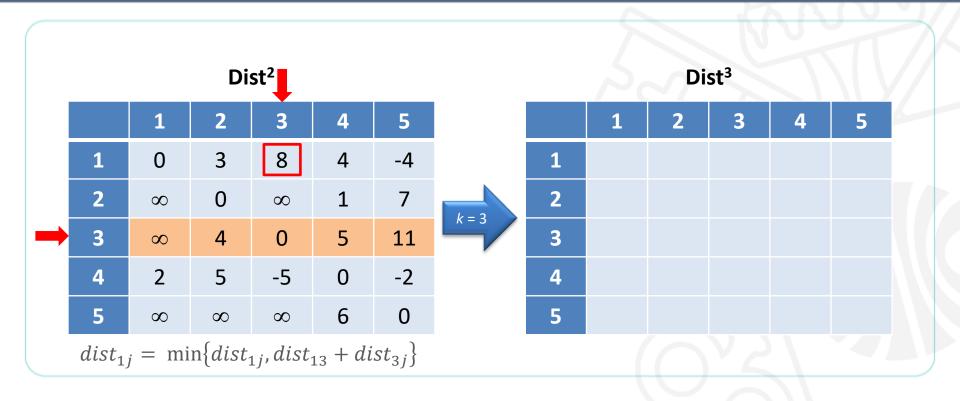


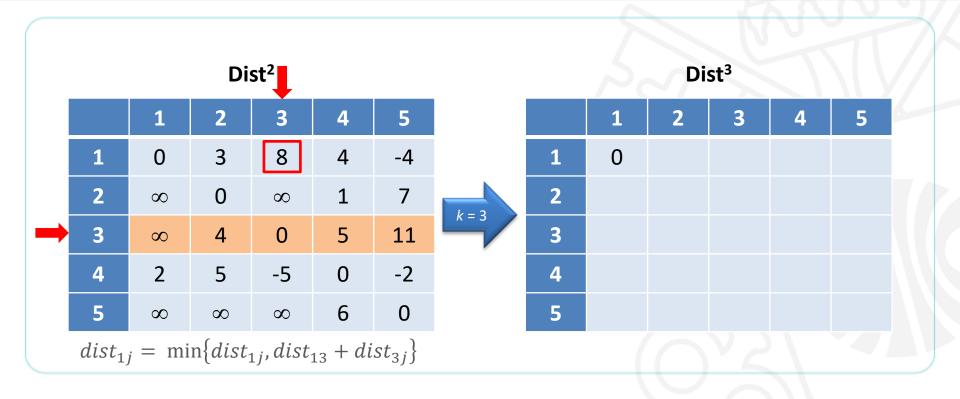


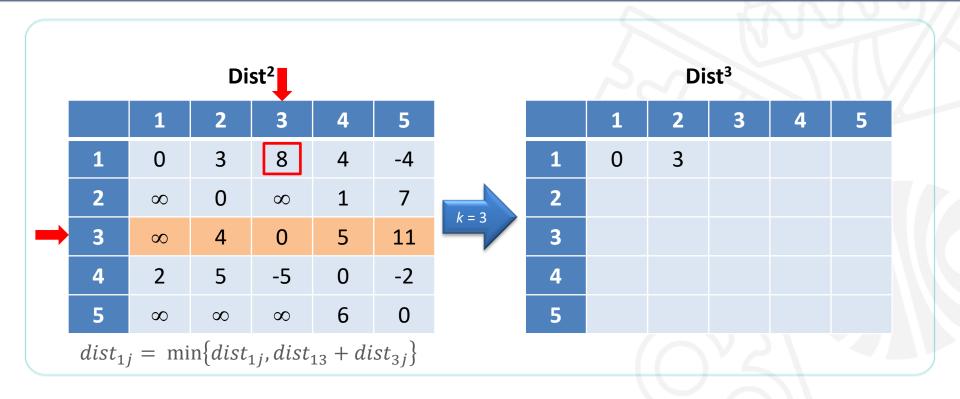


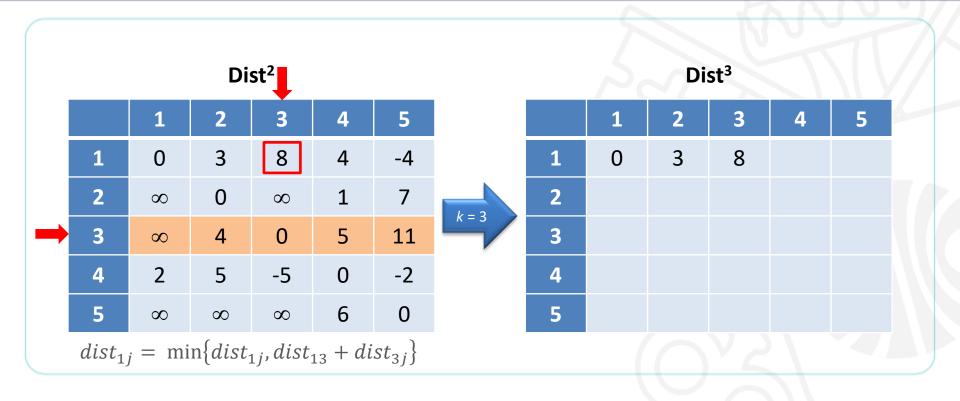


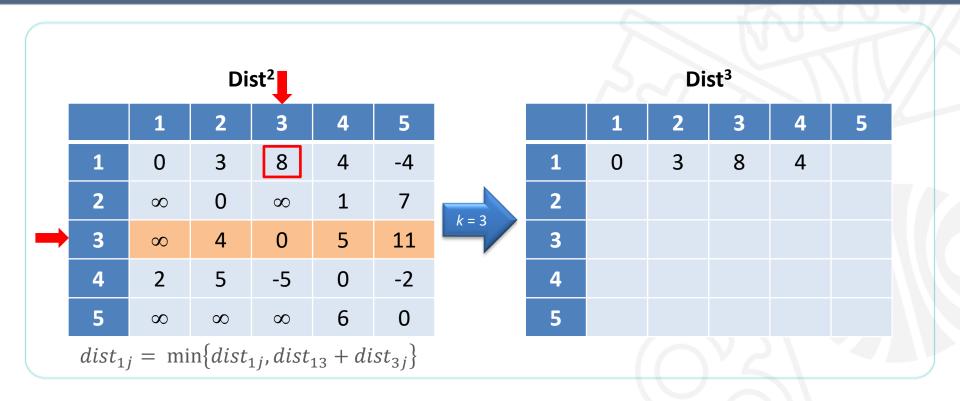


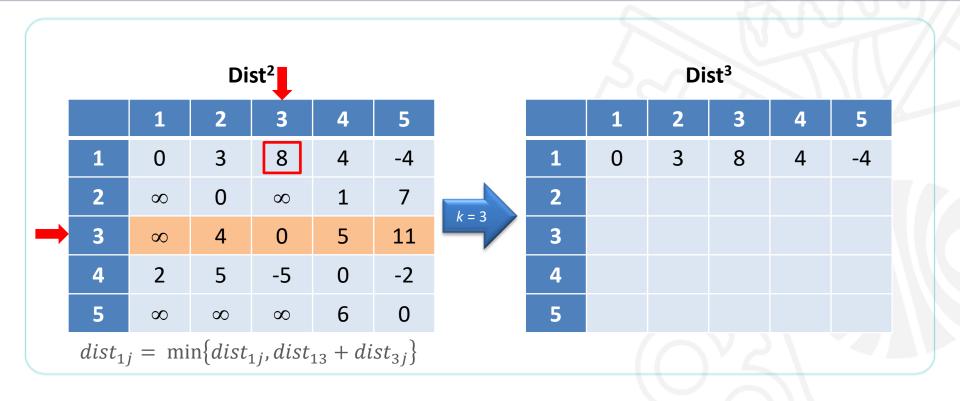


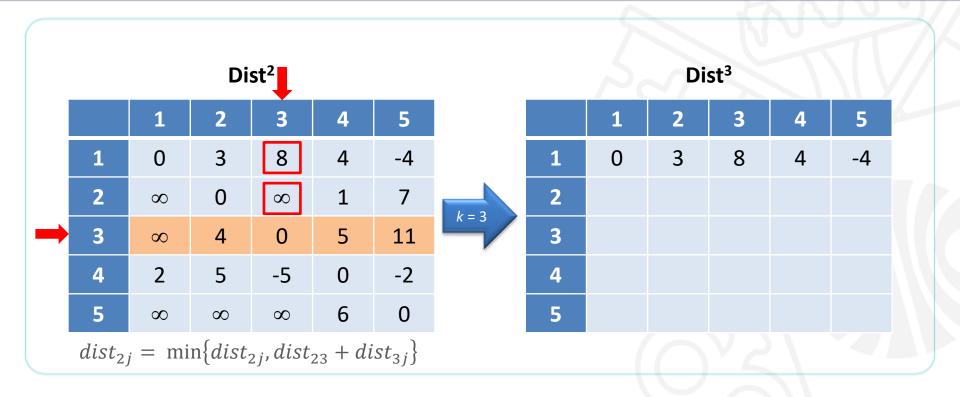


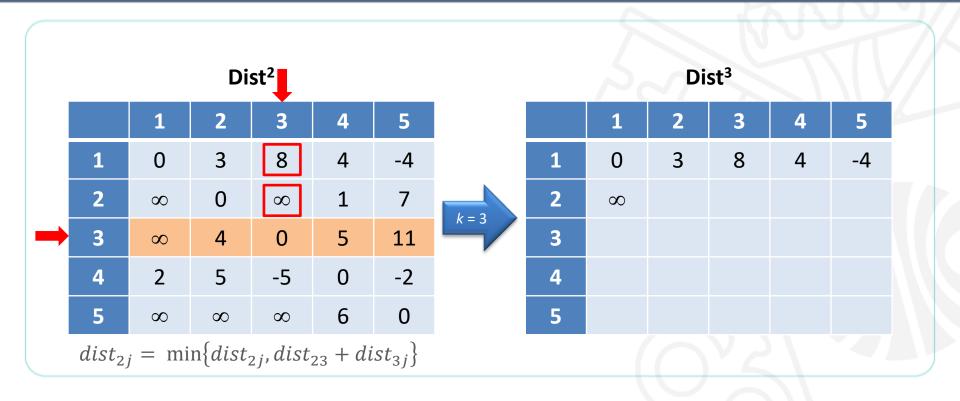


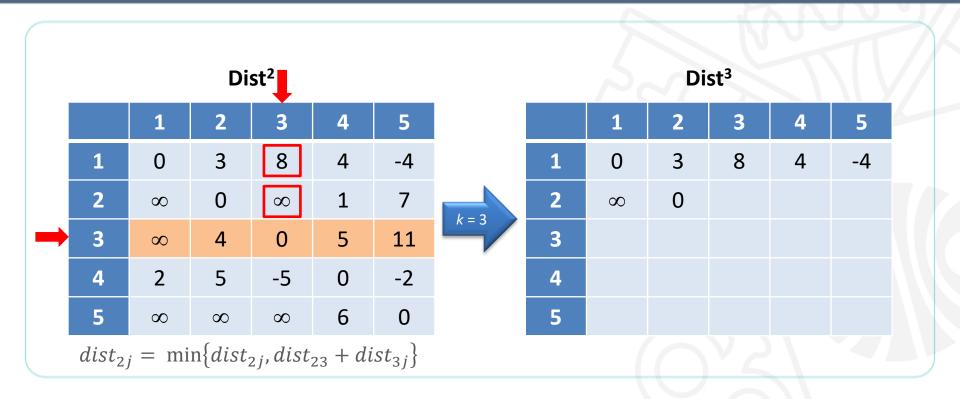


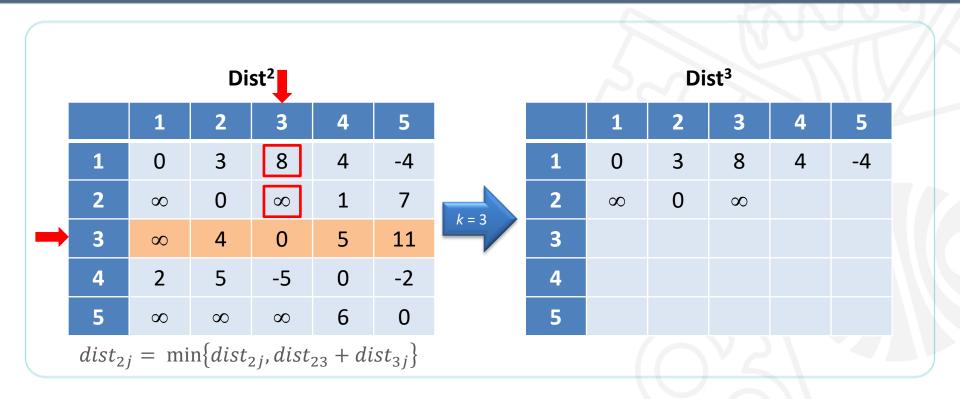


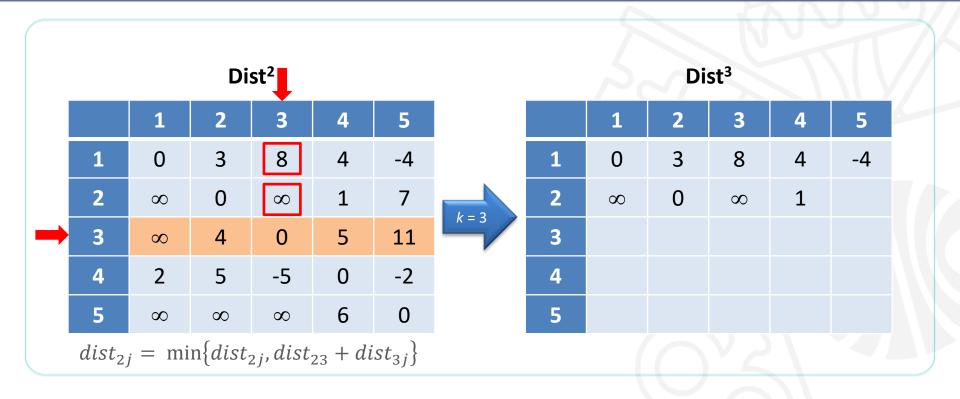


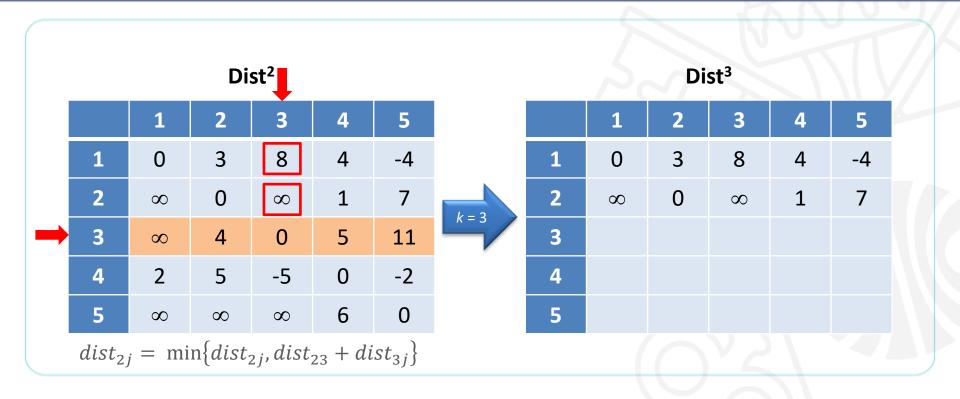


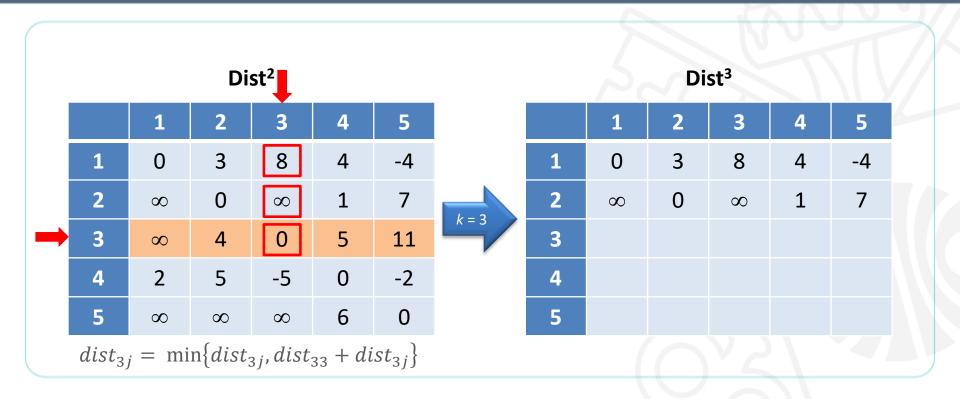


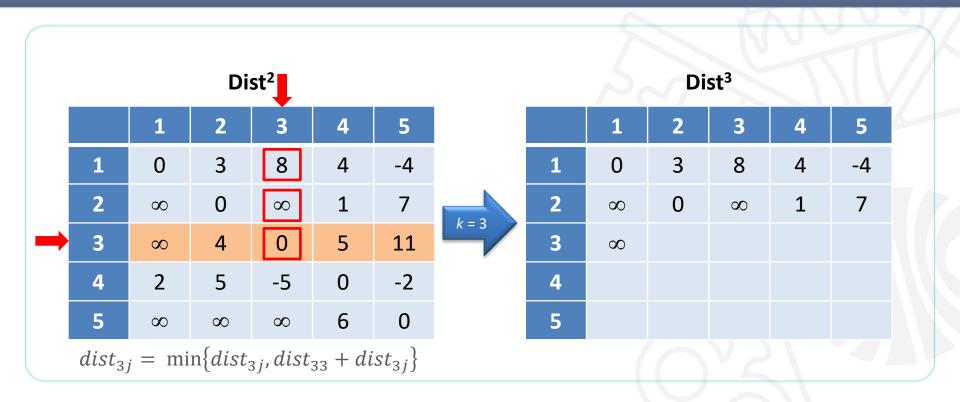


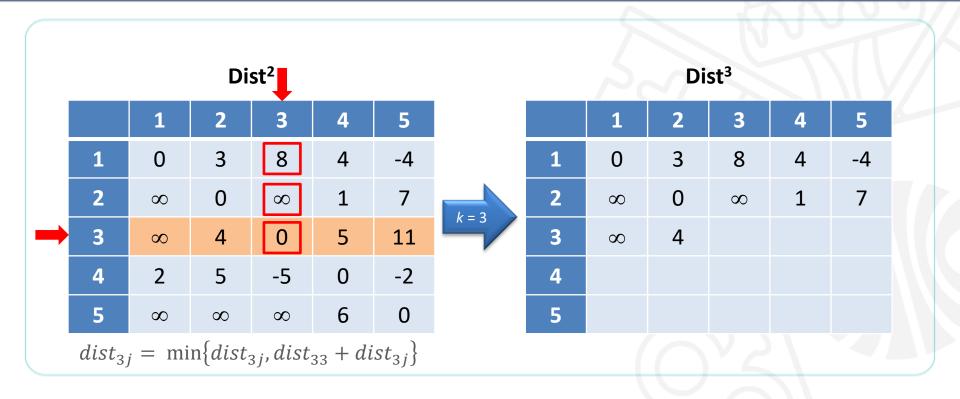


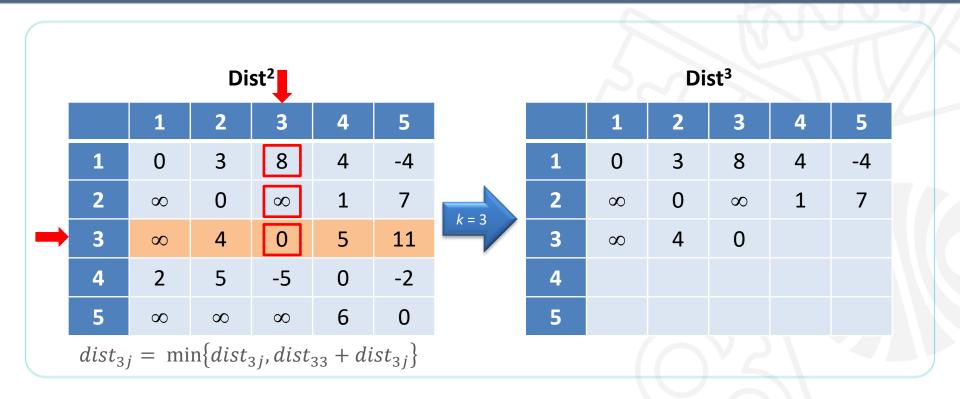


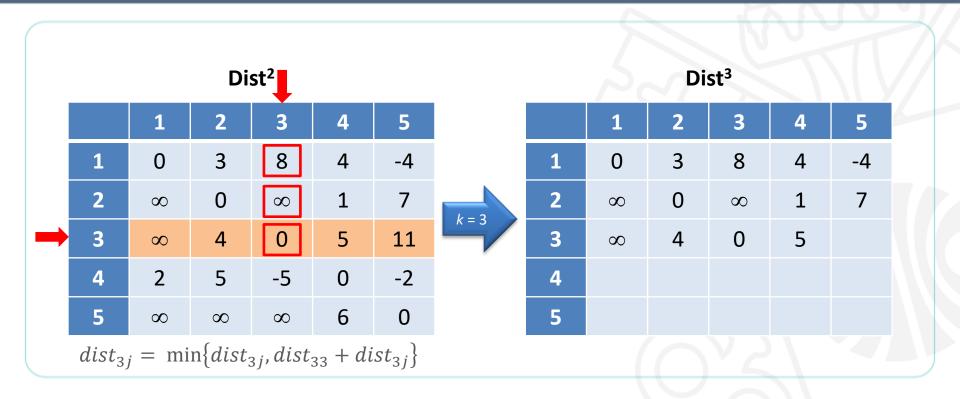


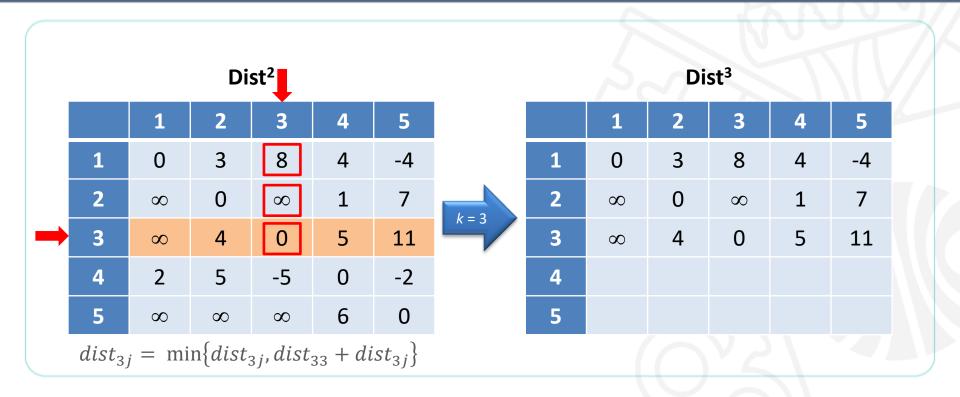


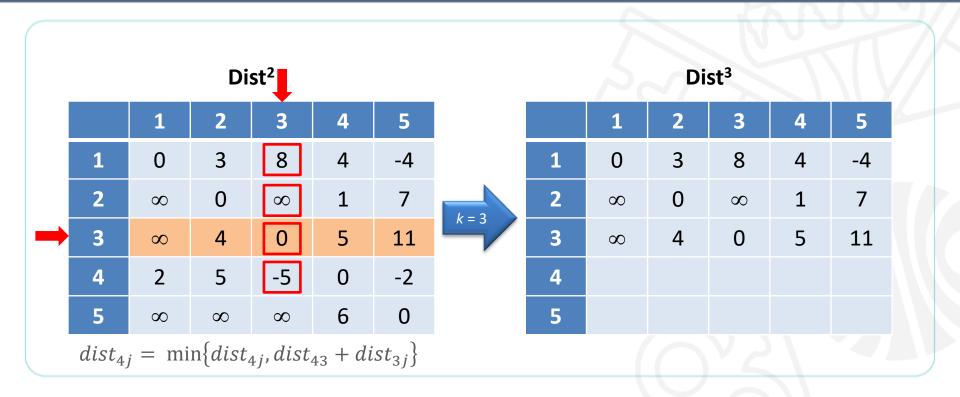


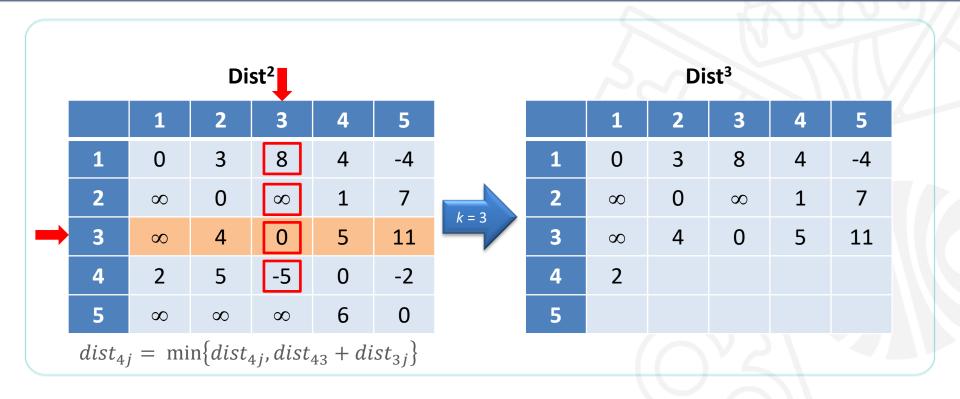


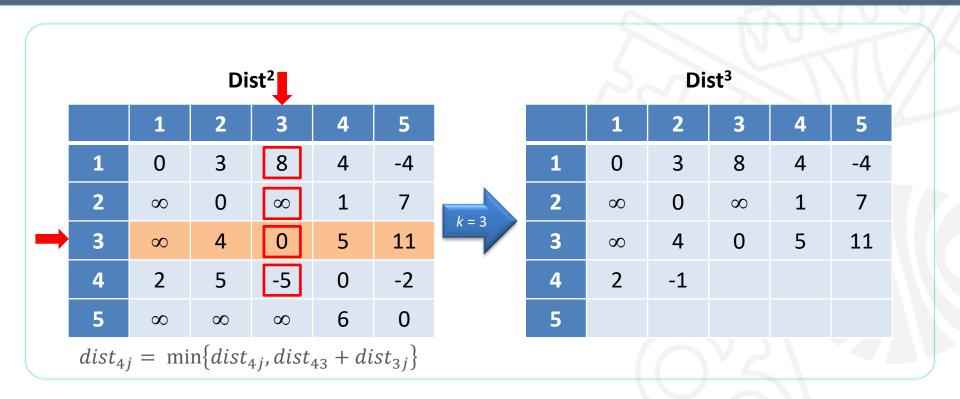


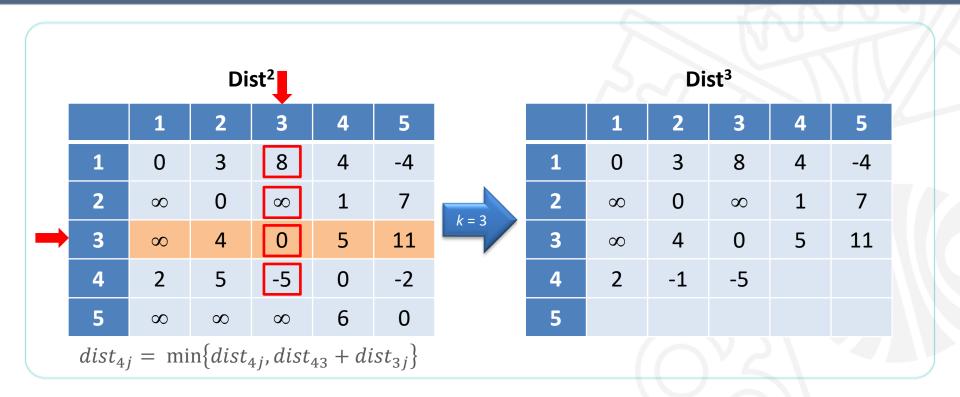


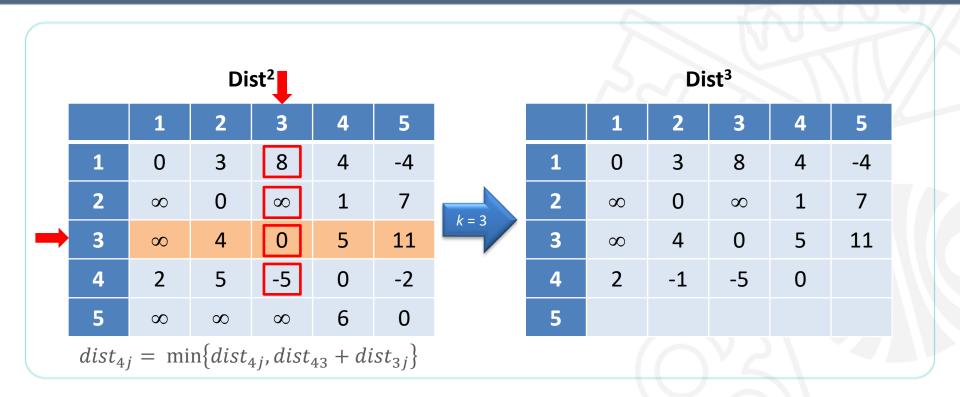


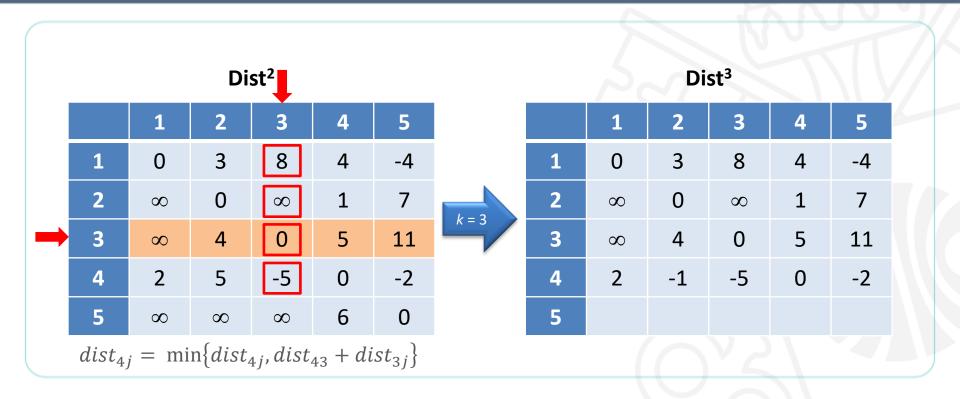


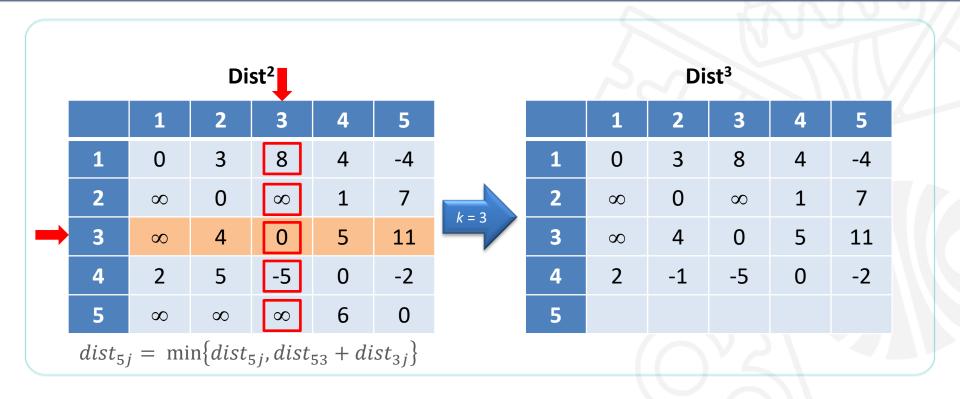


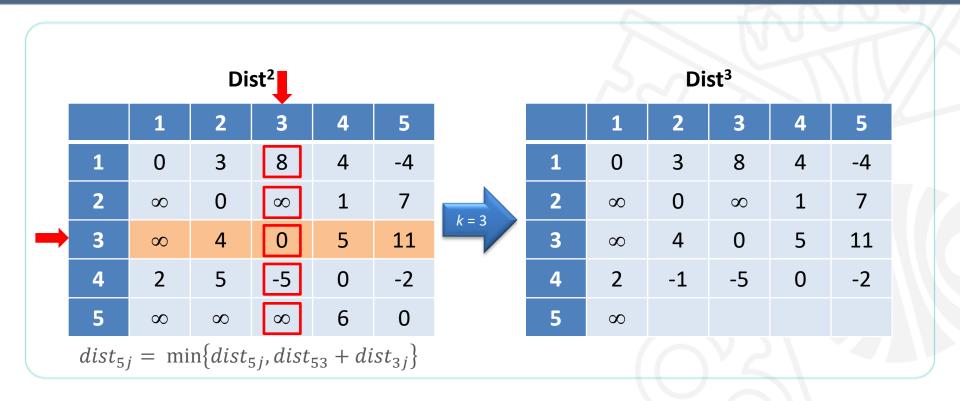


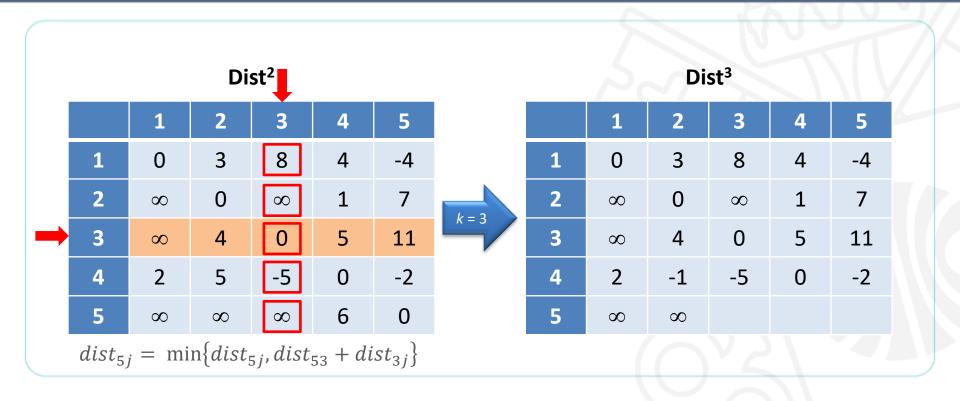


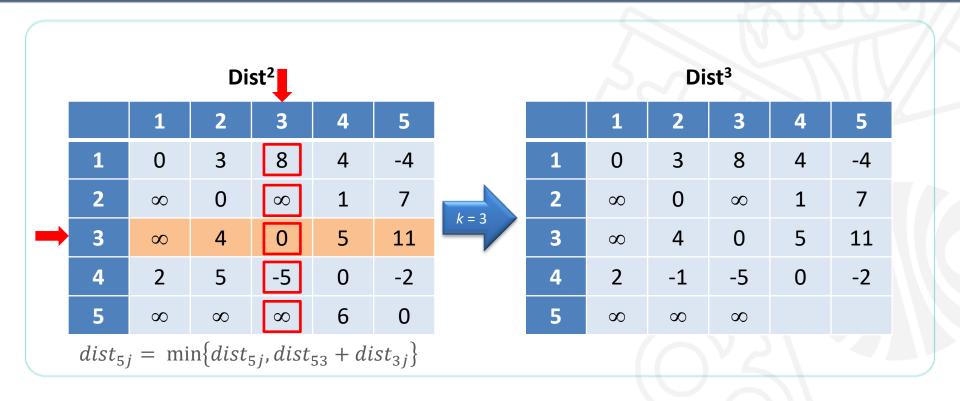


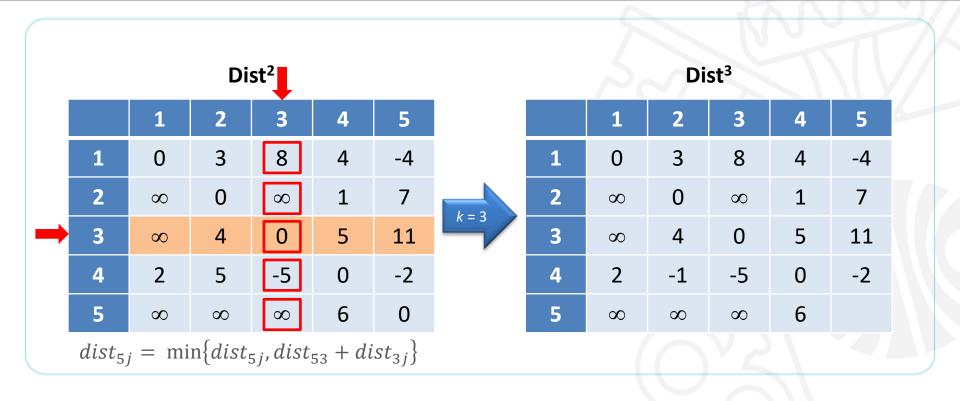


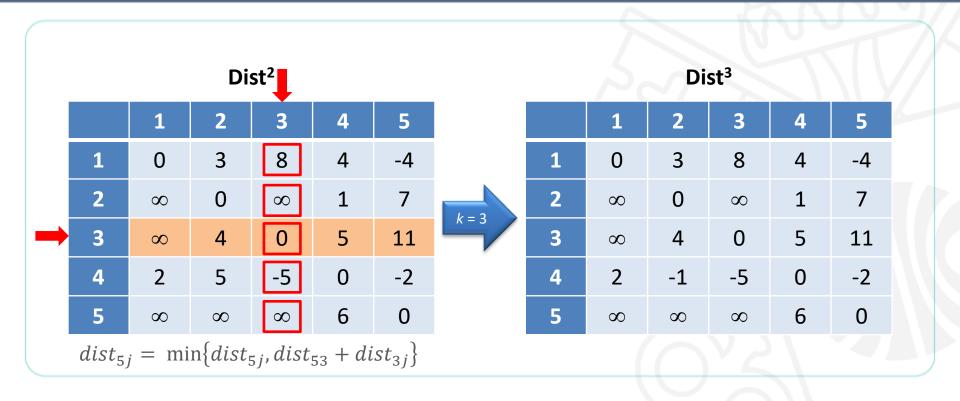


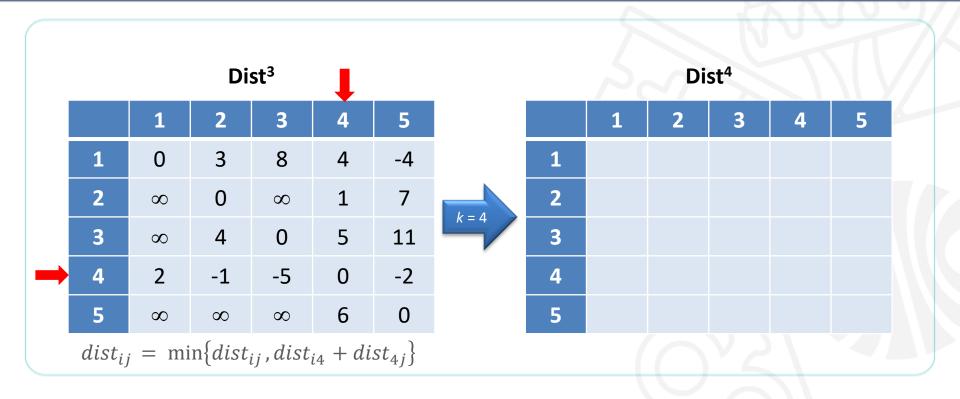


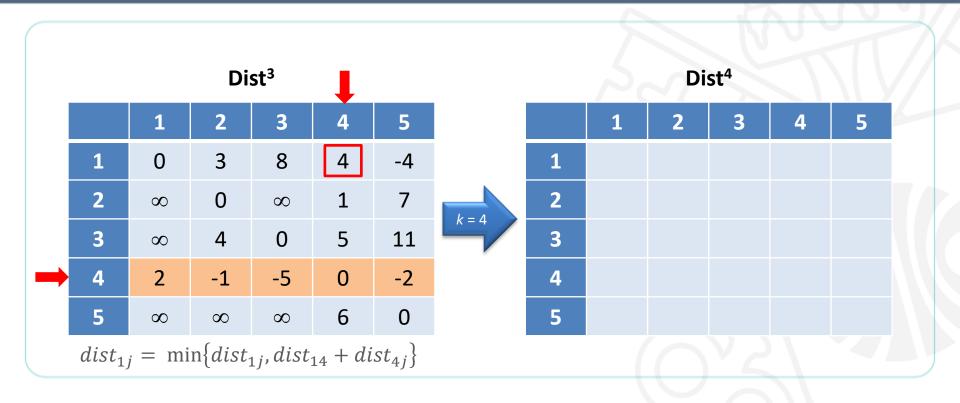


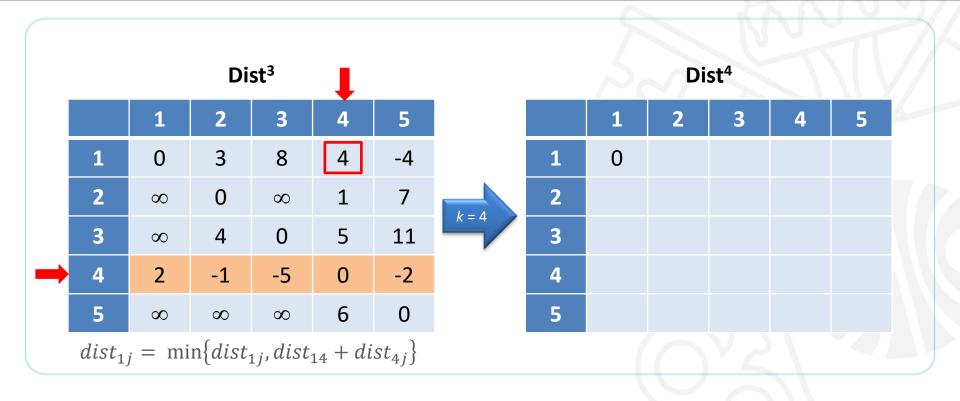


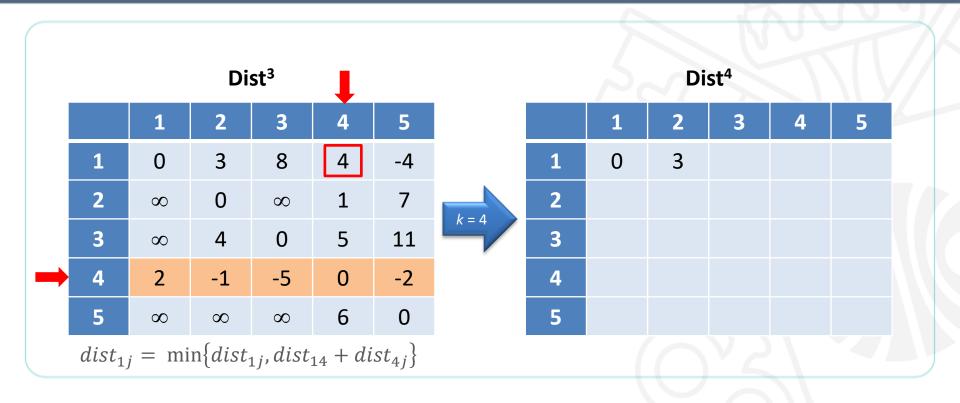


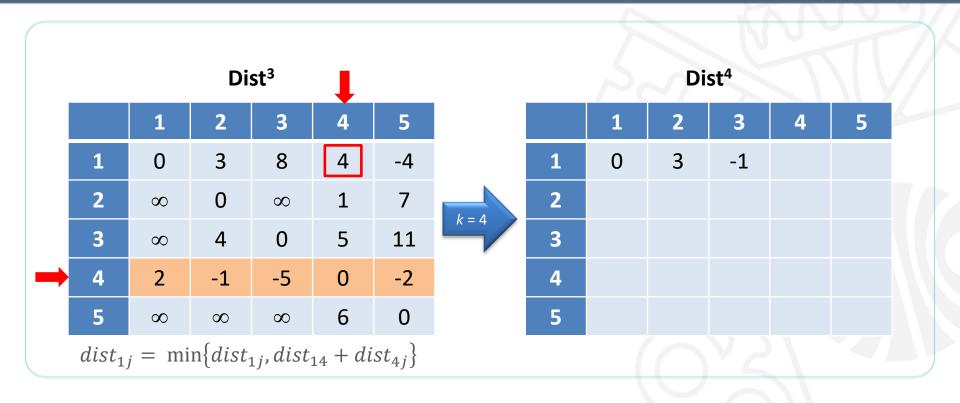


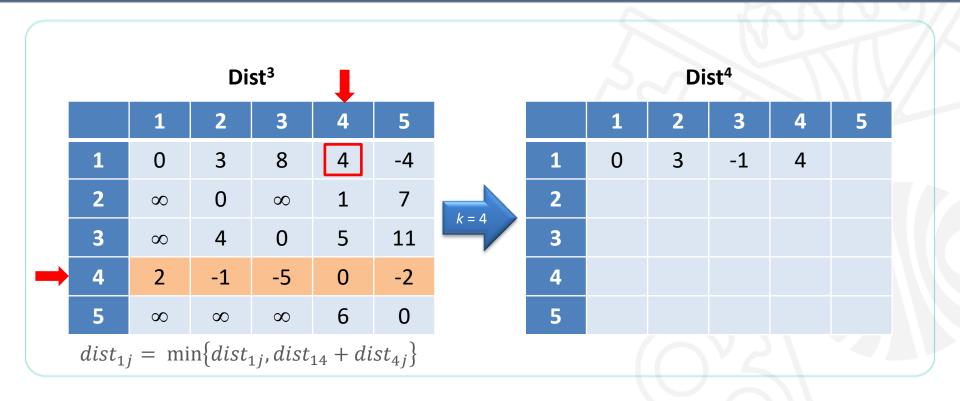


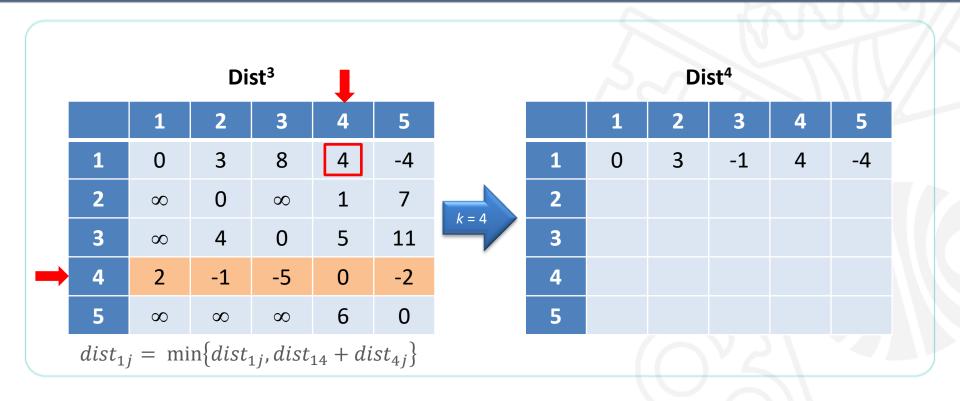


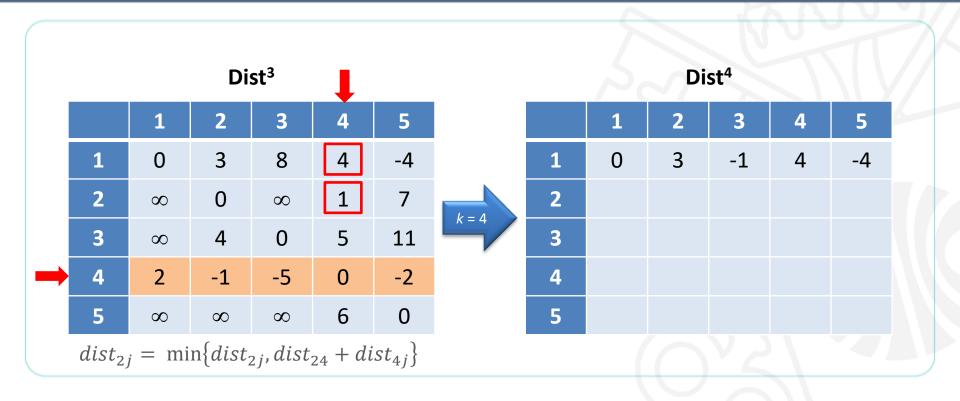


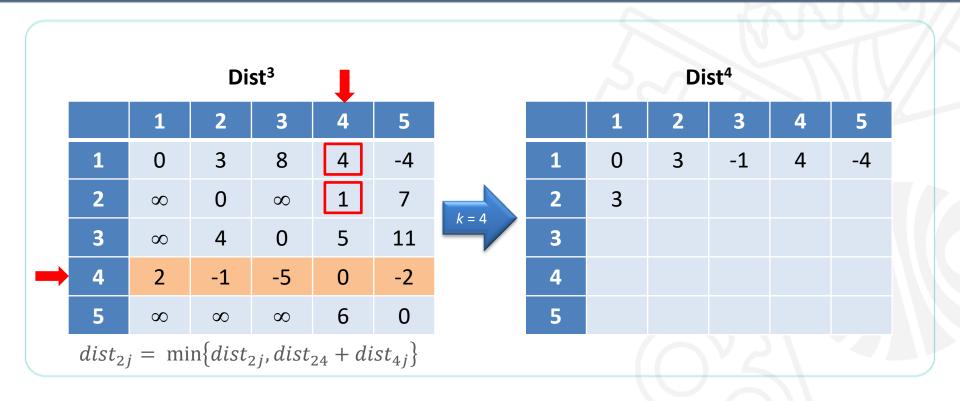


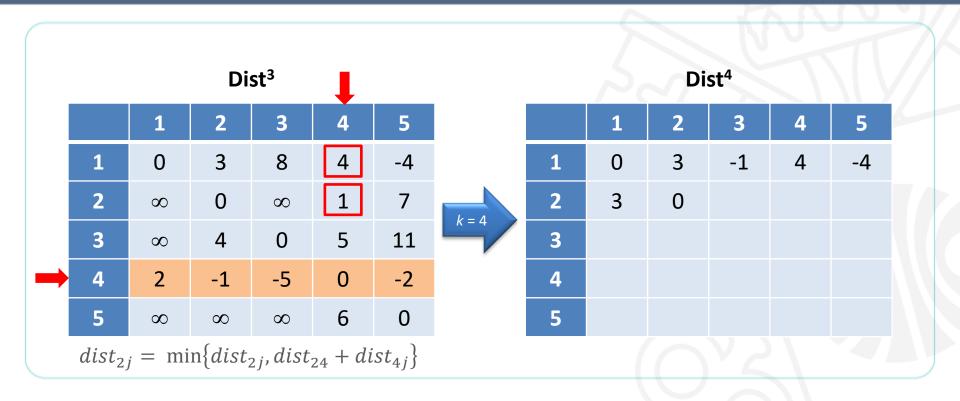


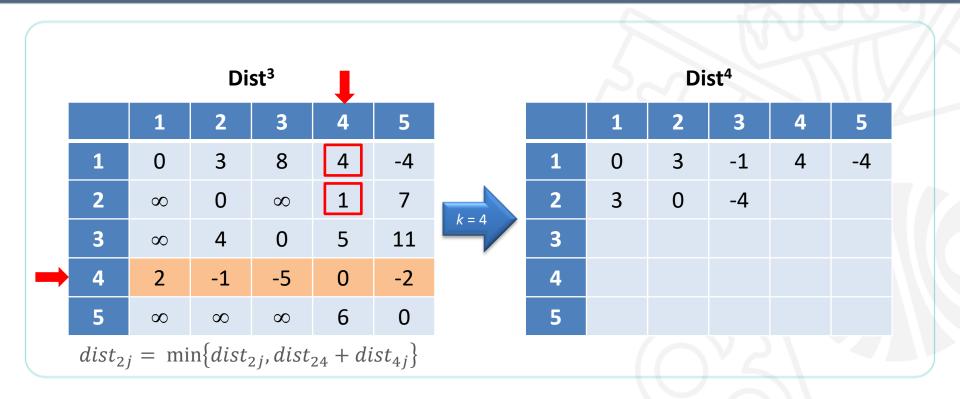


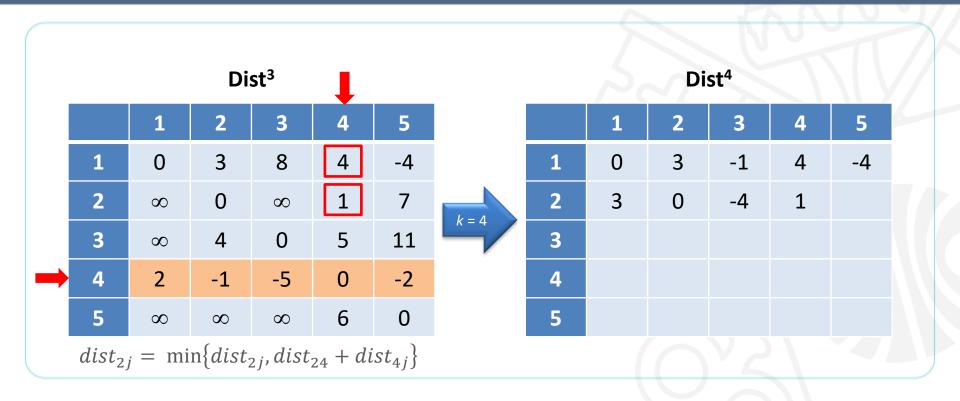


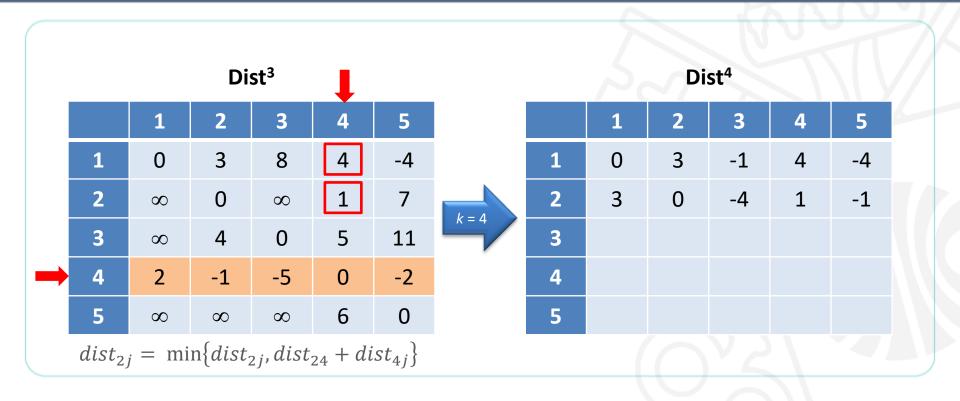


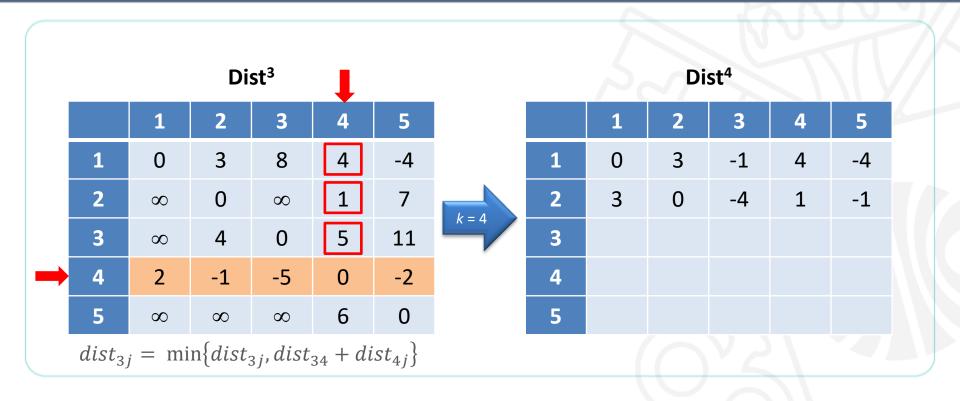


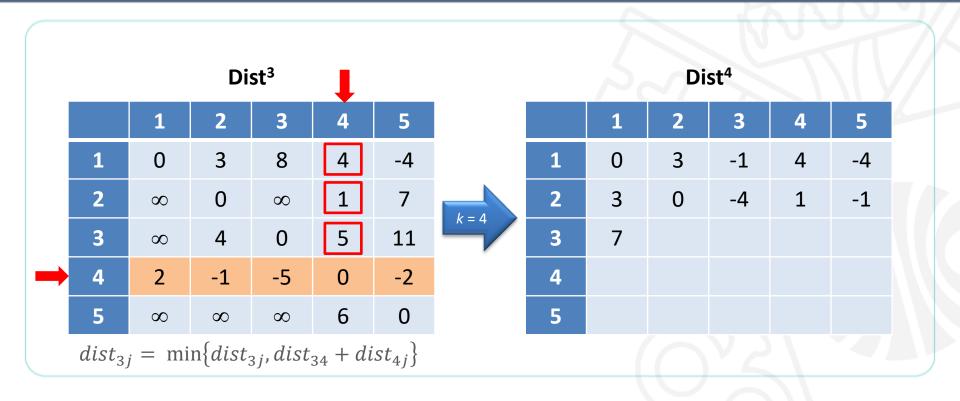


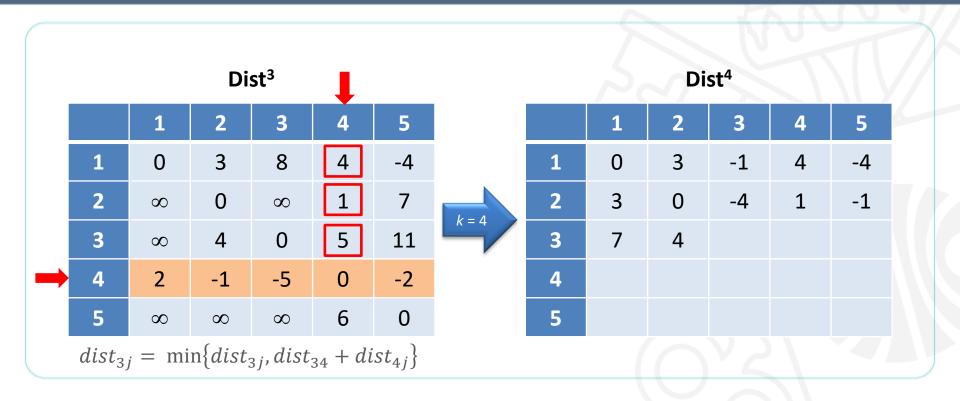


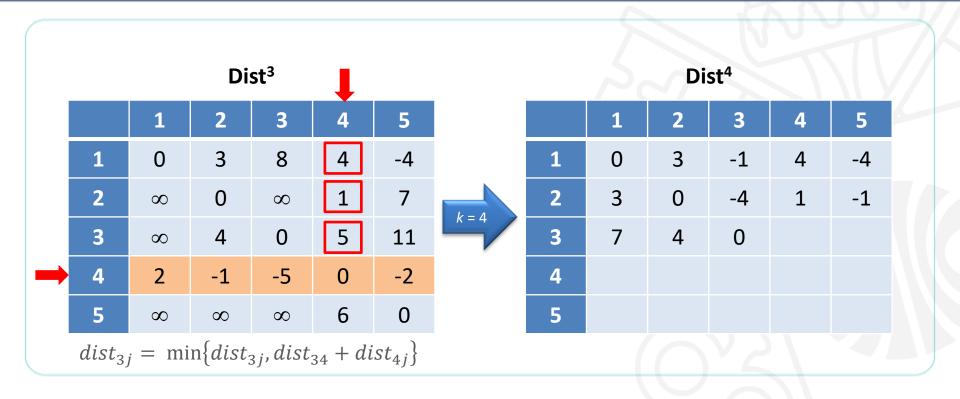


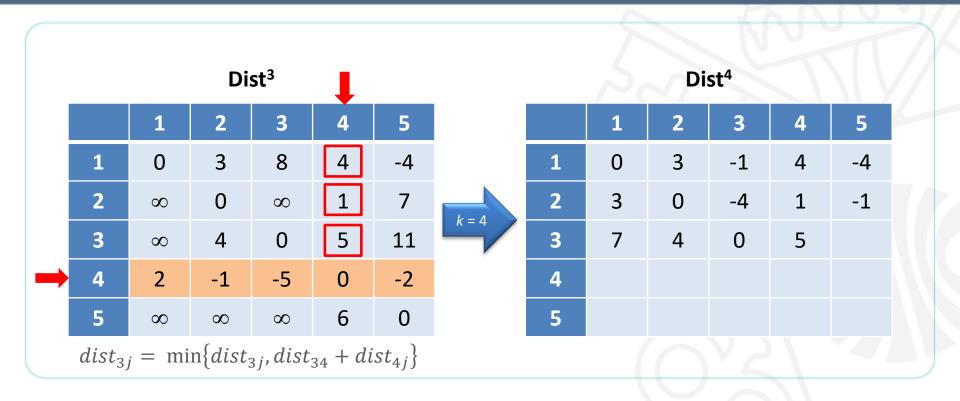


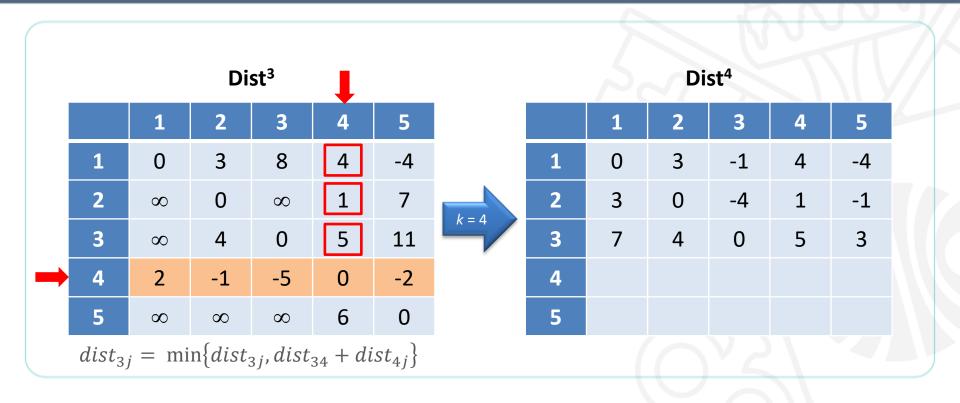


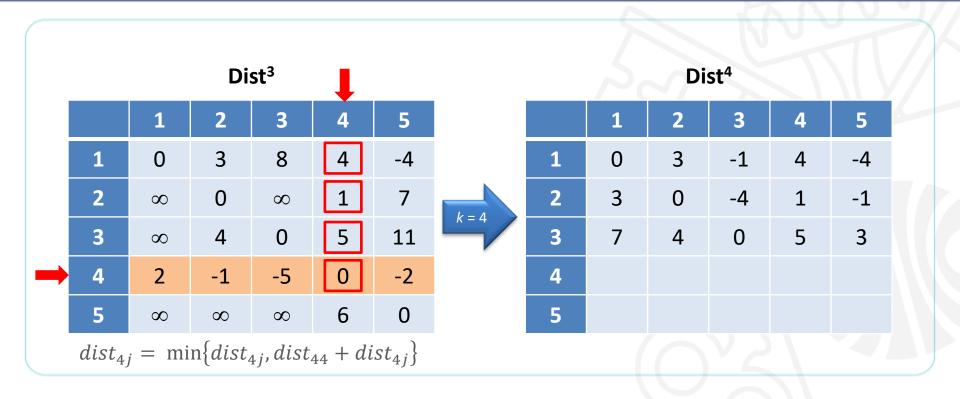


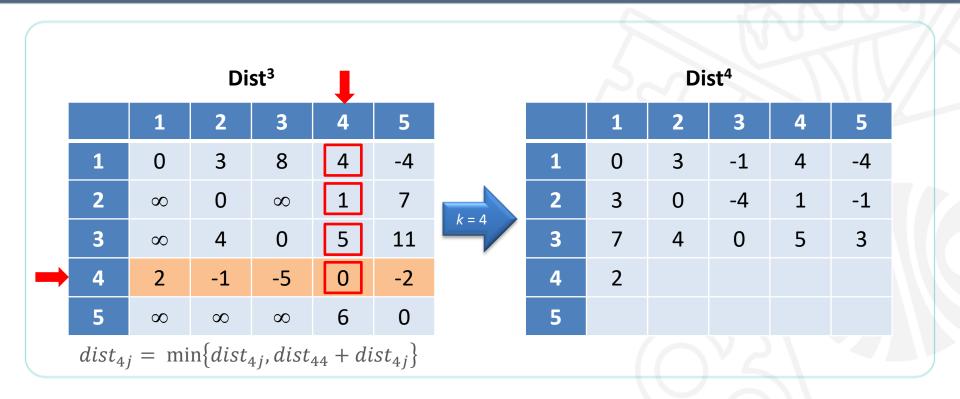


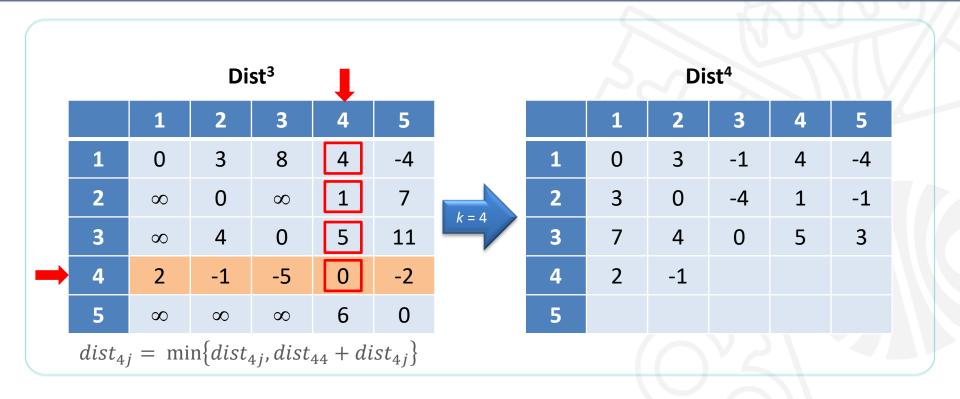


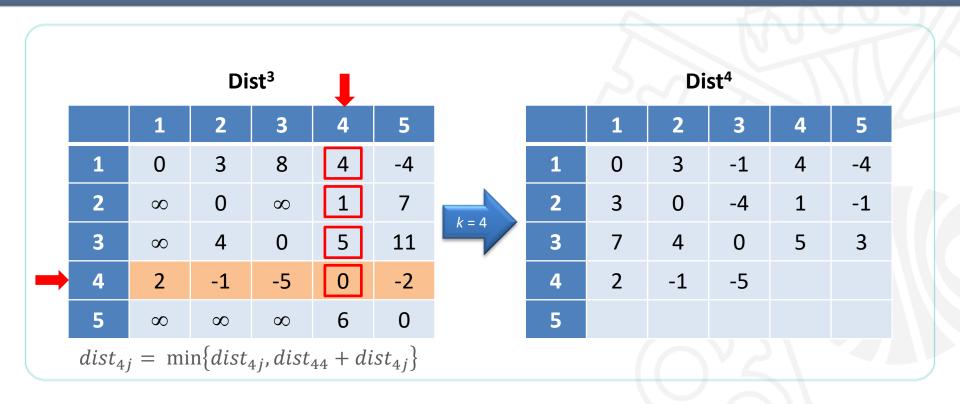


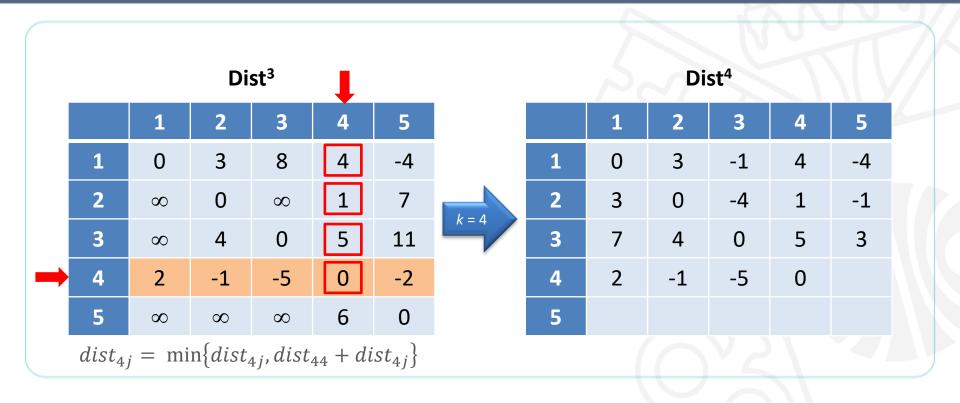


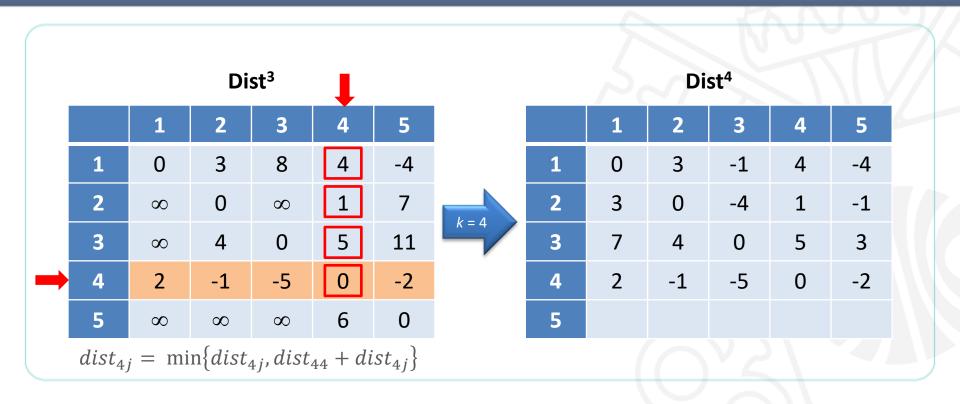


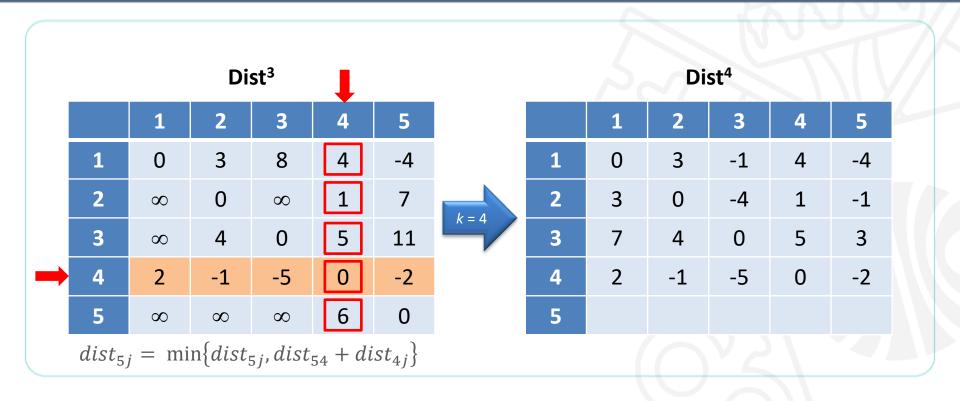


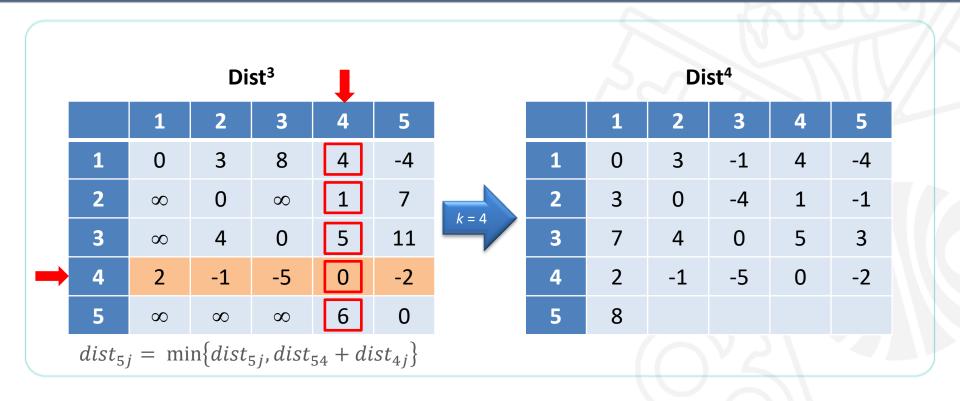


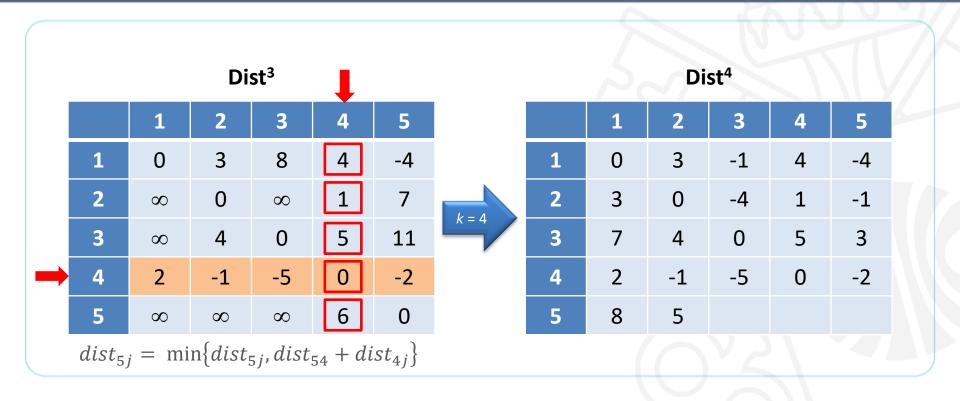


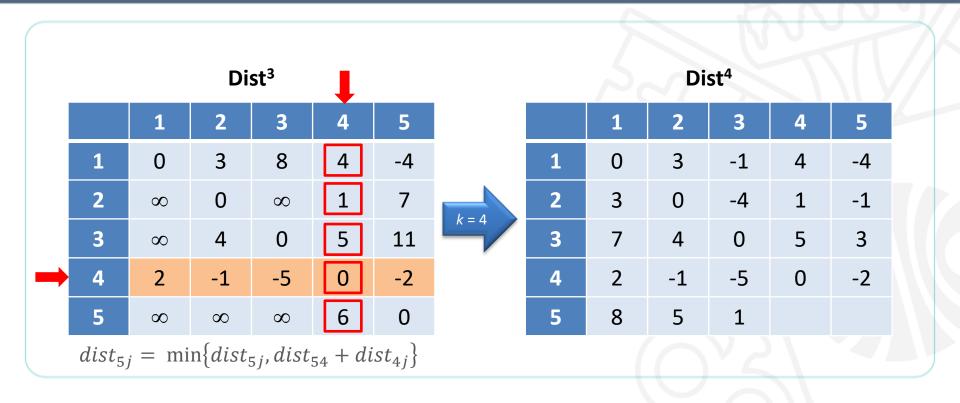


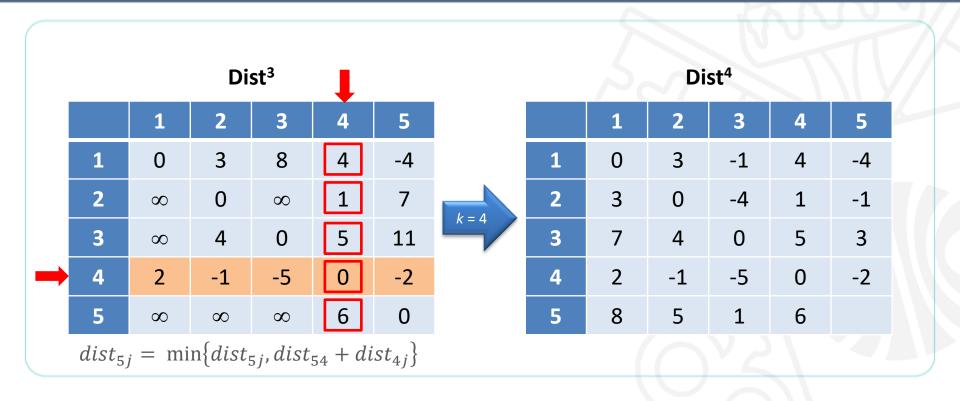


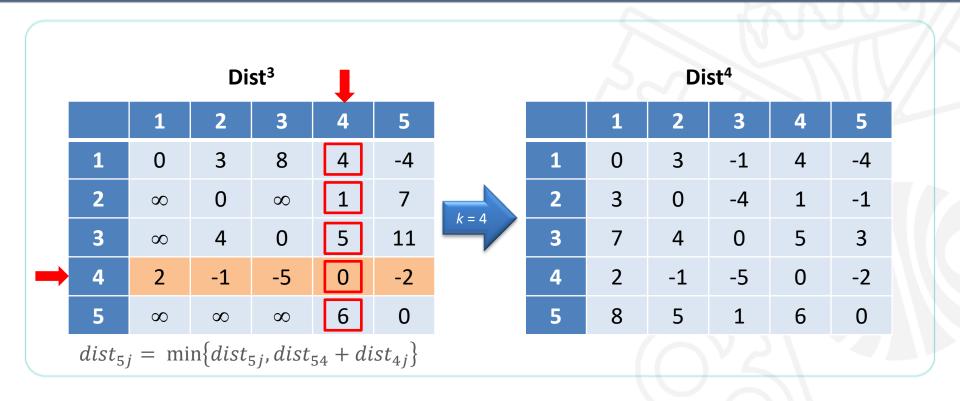


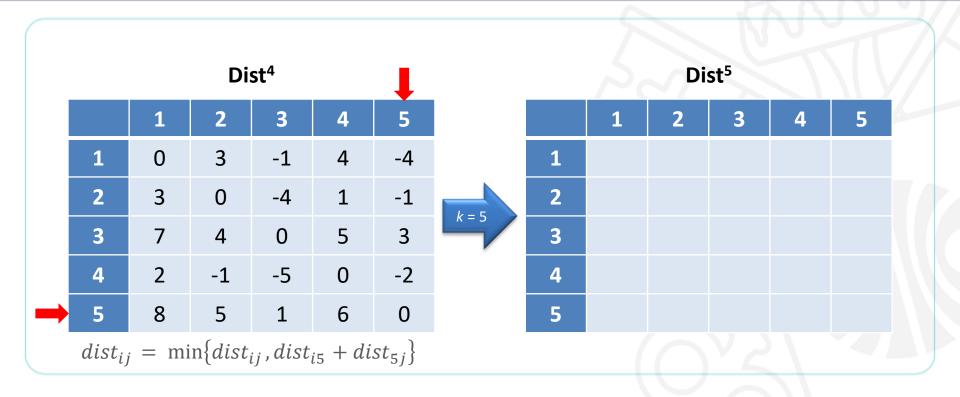


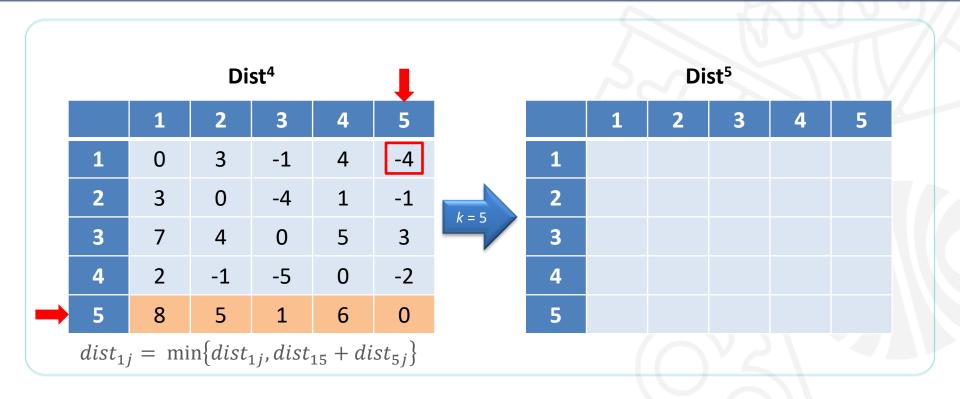


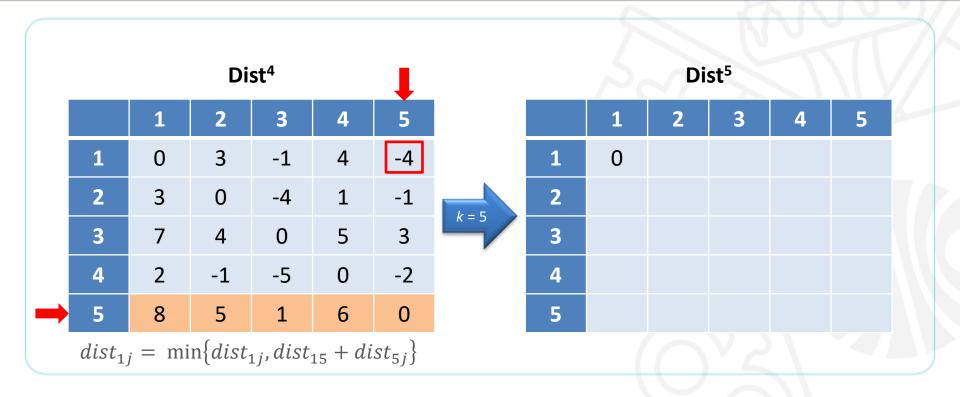


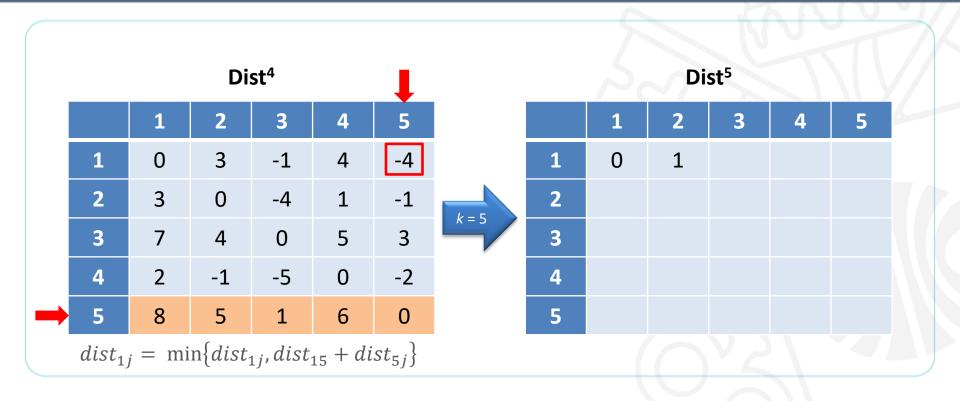


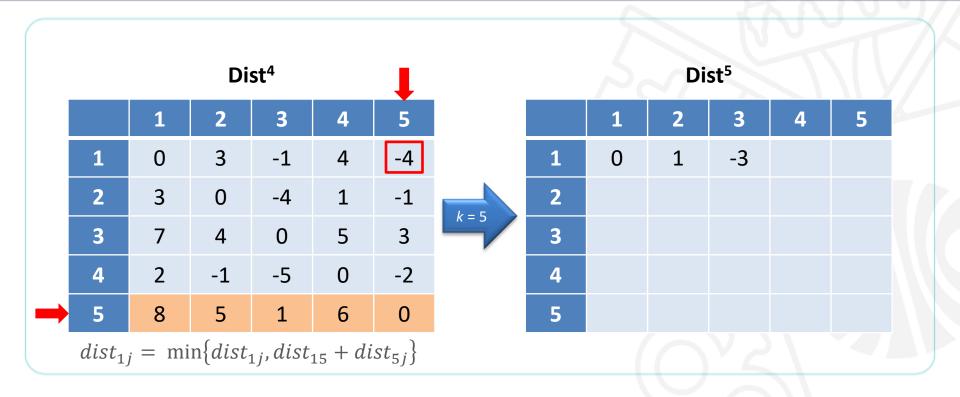


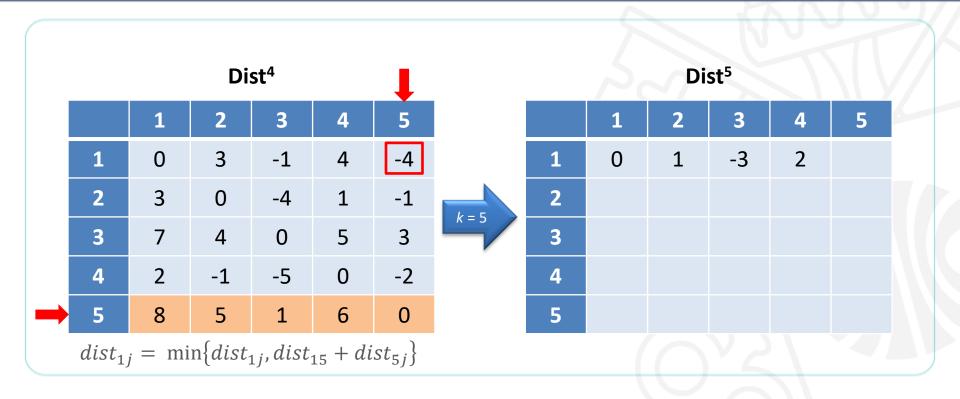


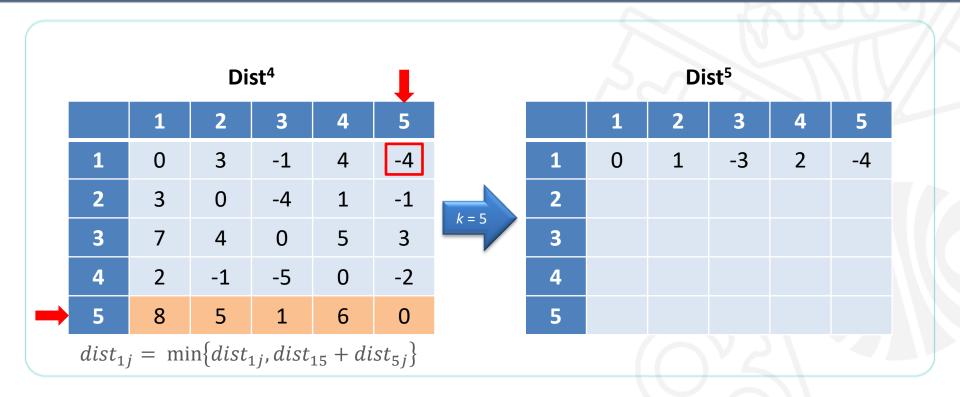


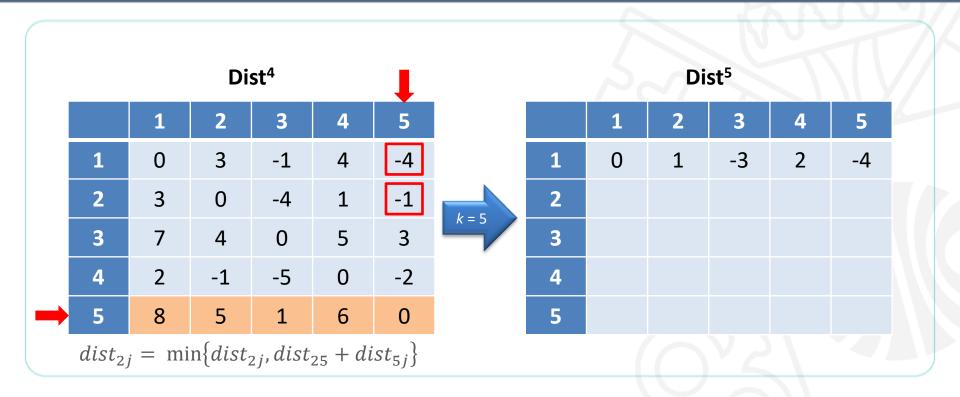


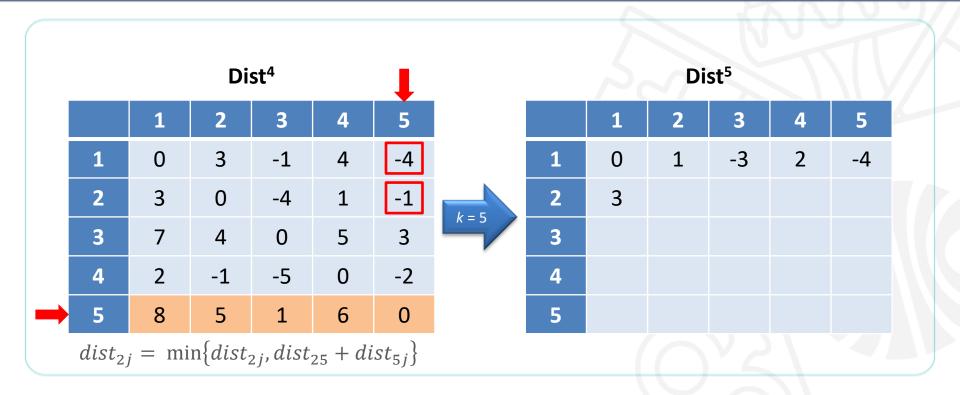


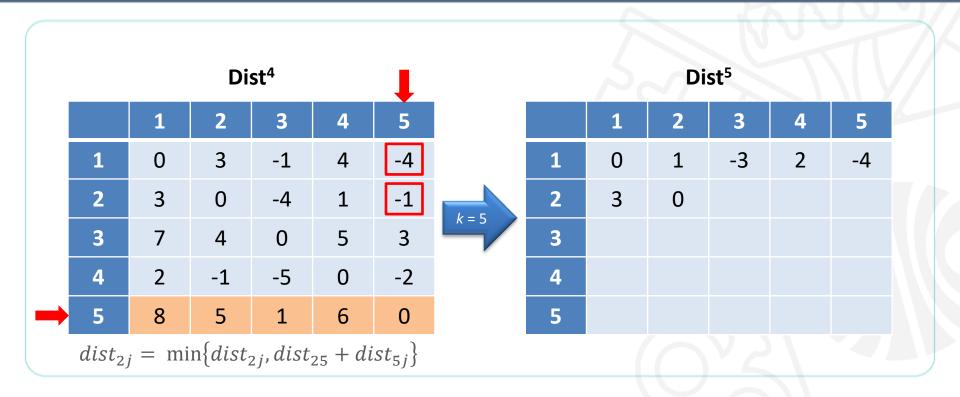


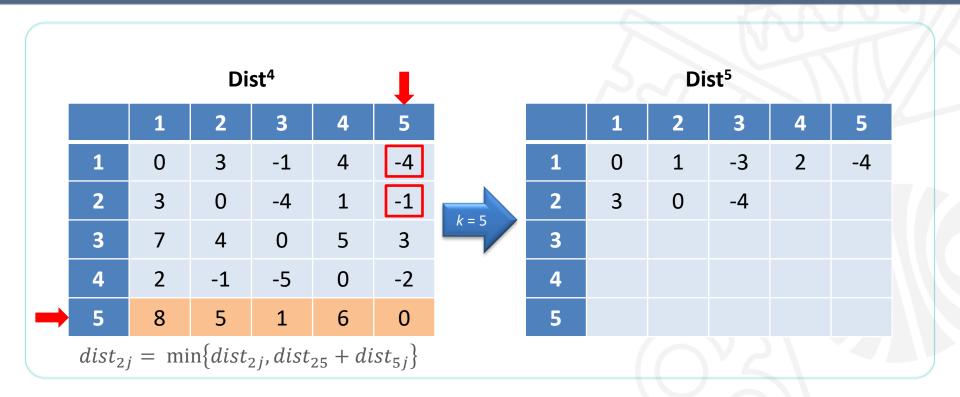


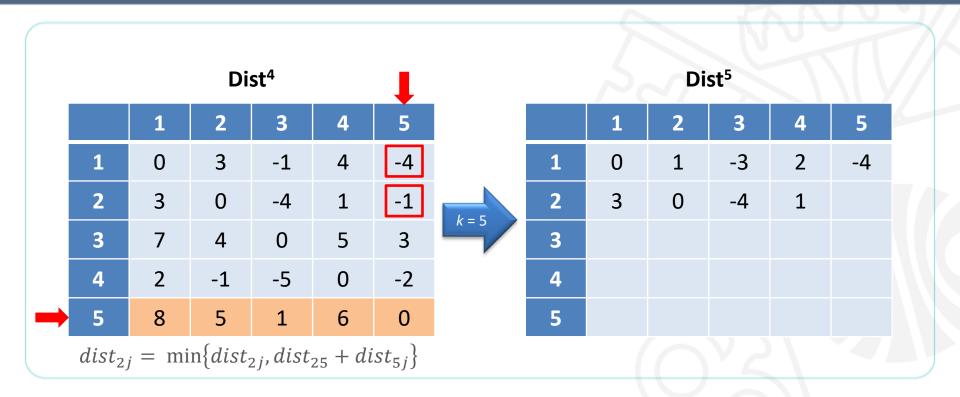


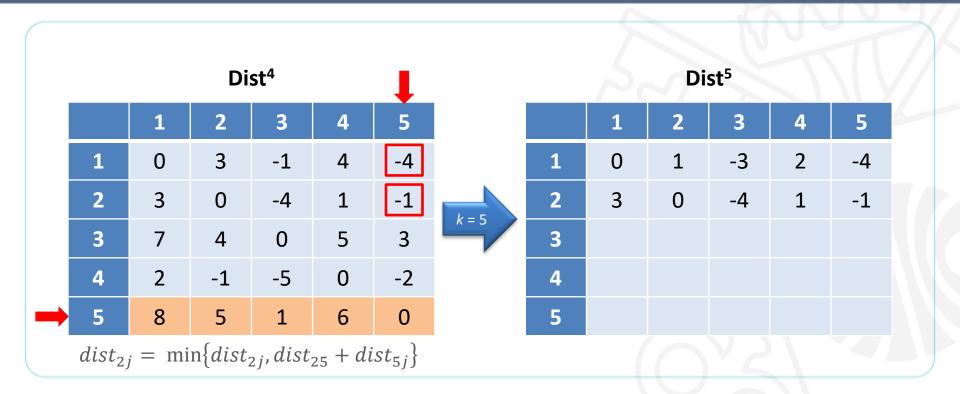


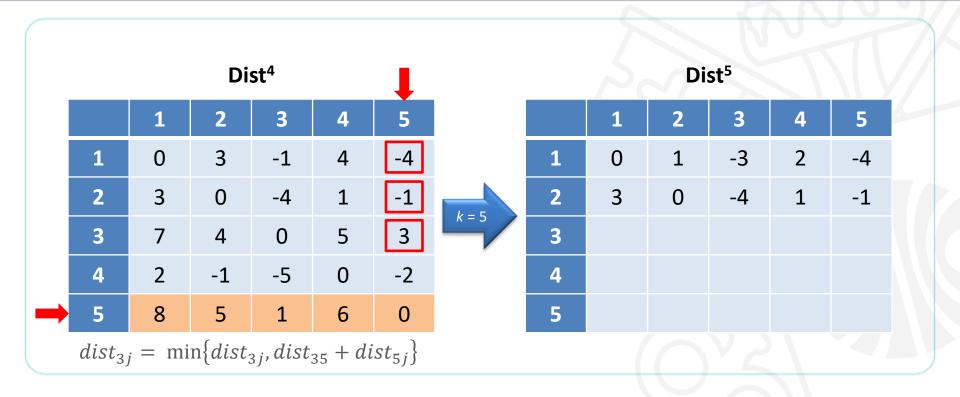


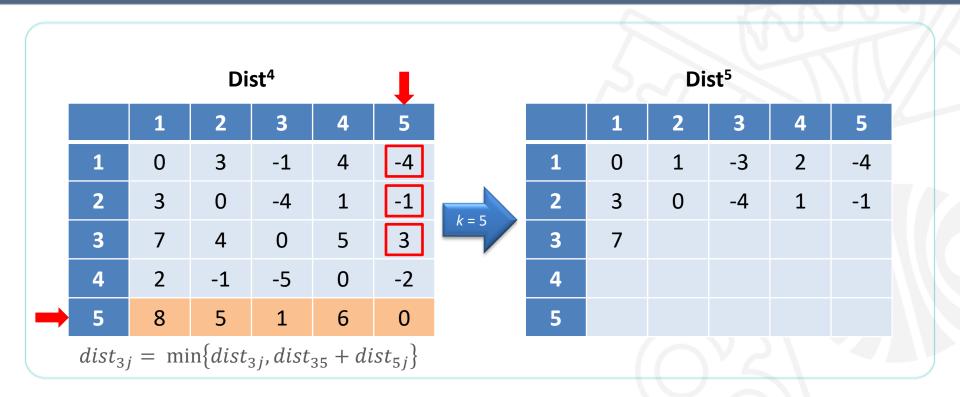


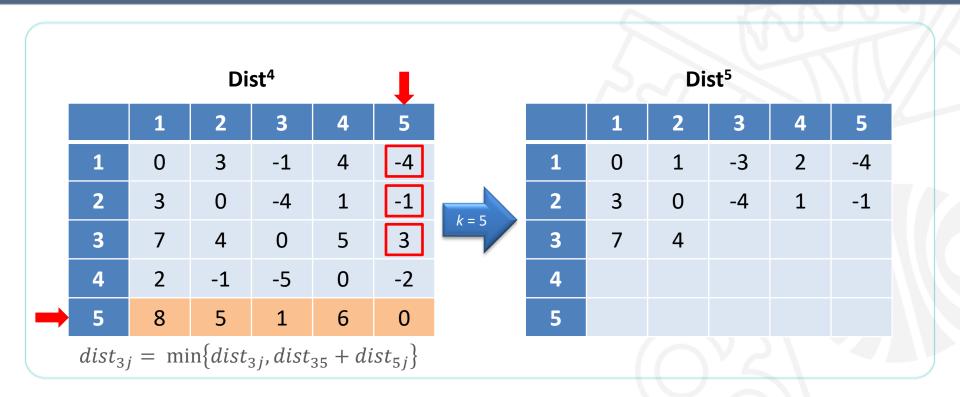


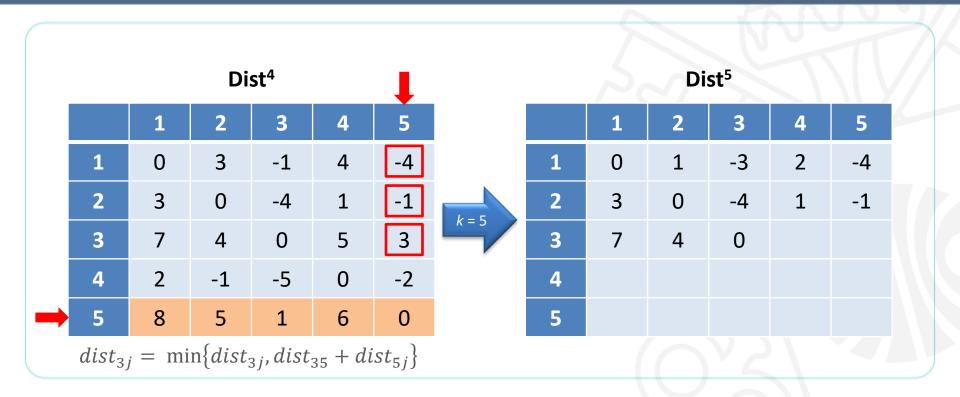


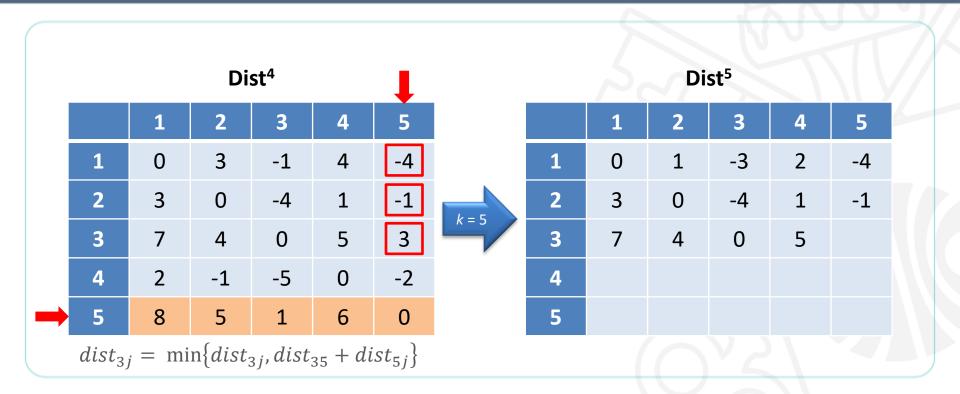


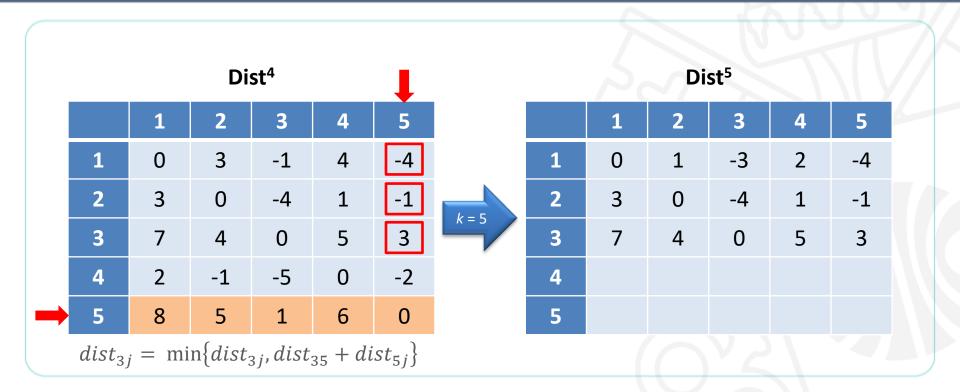


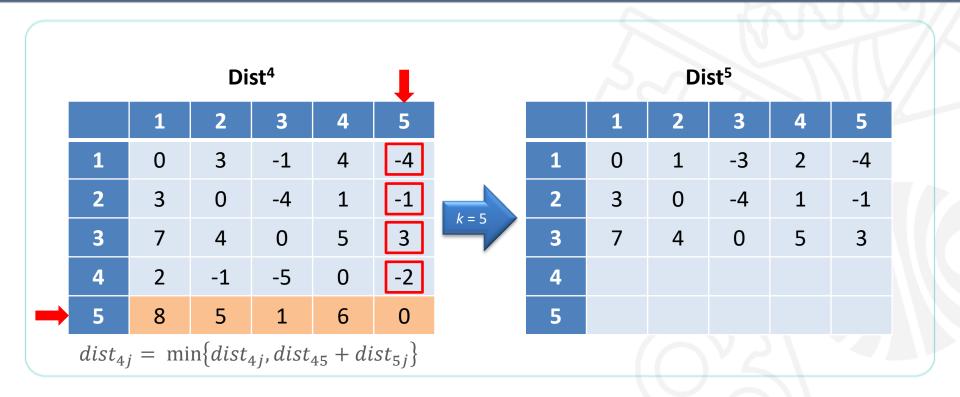


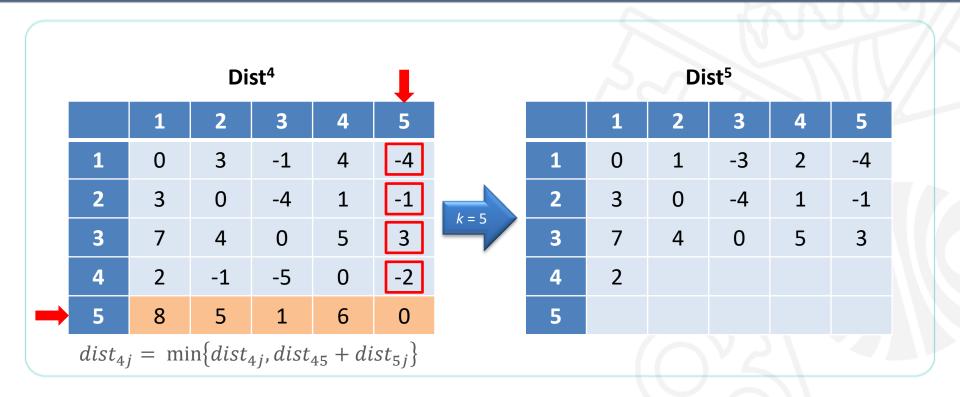


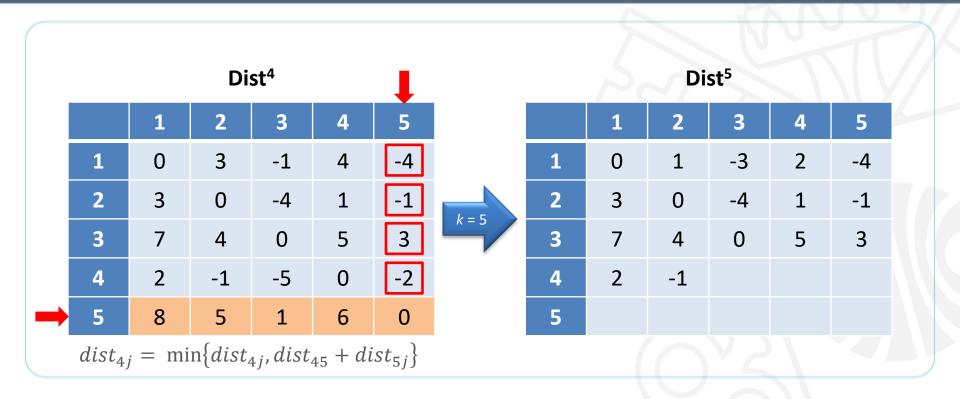


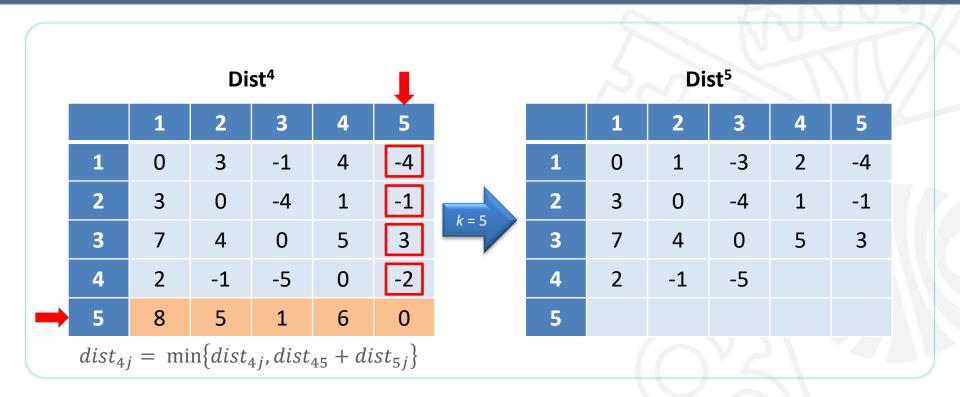


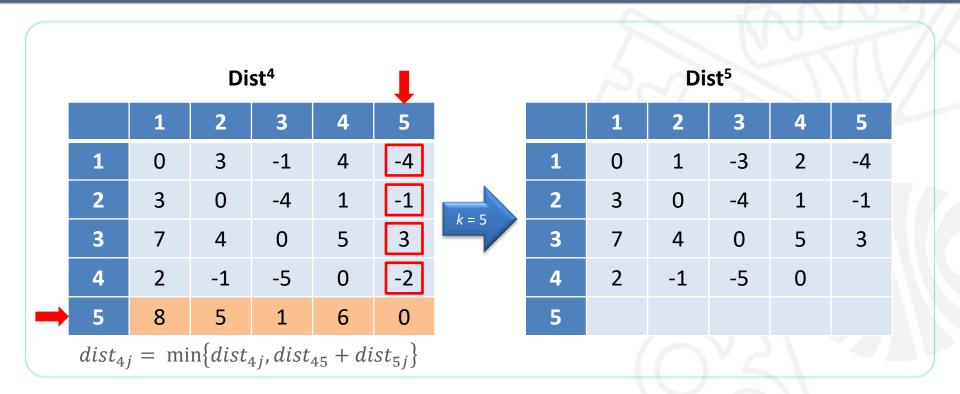


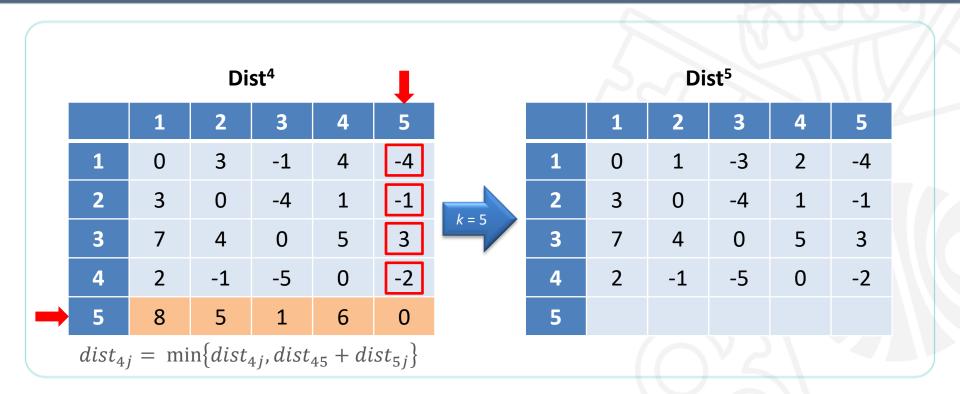


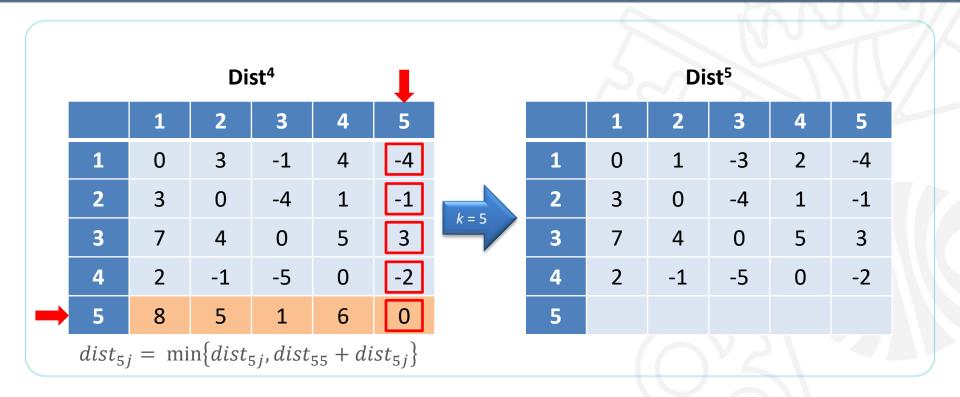


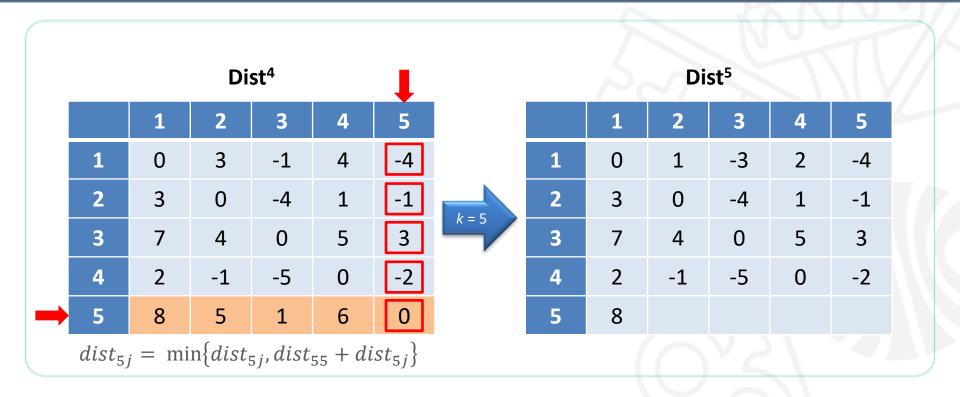


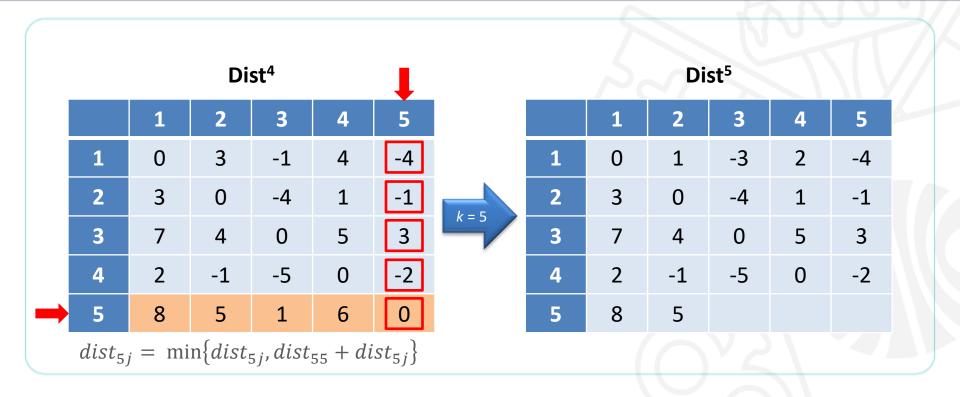


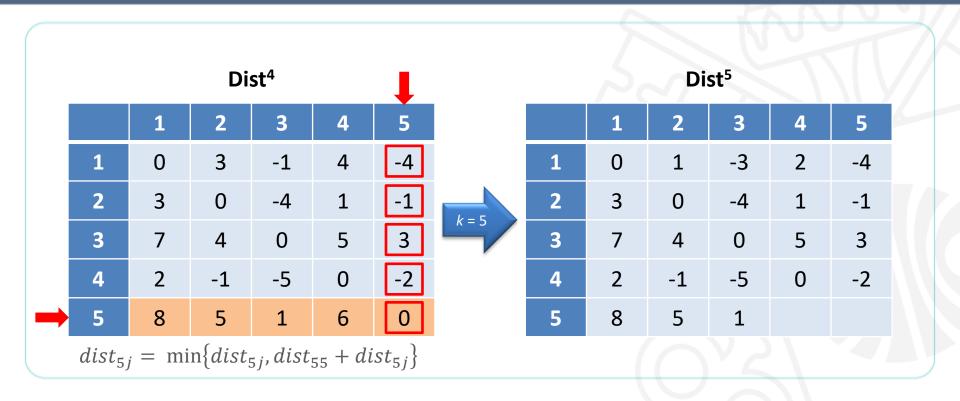


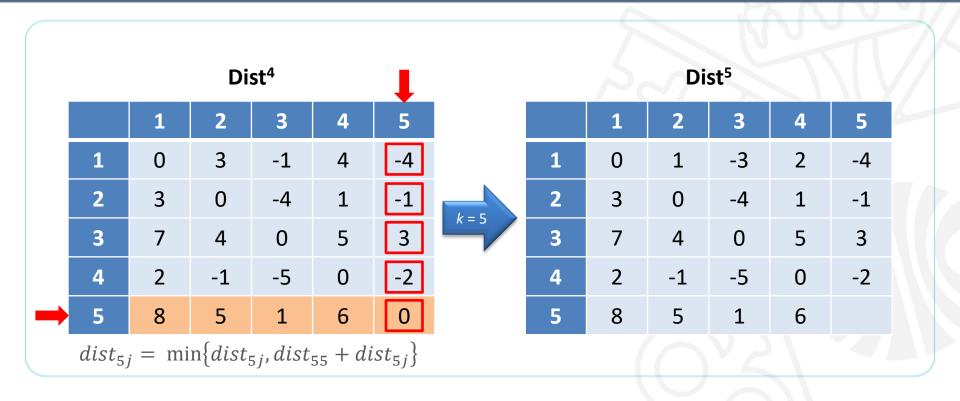


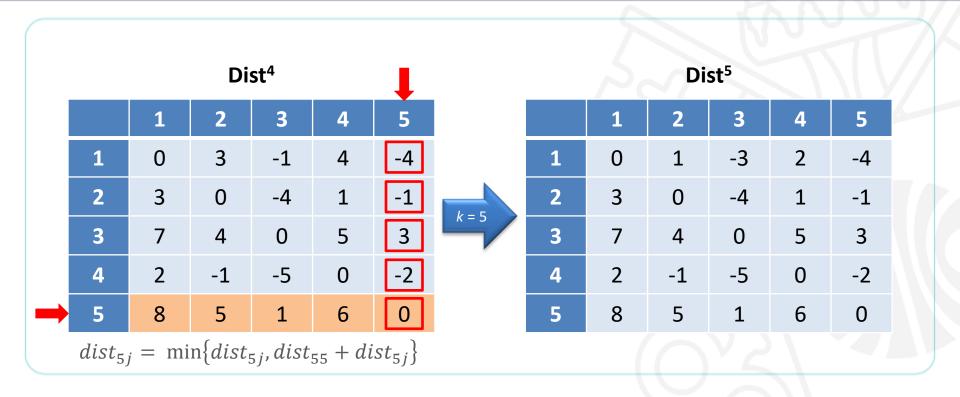


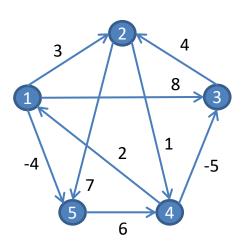












Distâncias Mínimas

	1	2	3	4	5
1	0	1	-3	2	-4
2	3	0	-4	1	-1
3	7	4	0	5	3
4	2	-1	-5	0	-2
5	8	5	1	6	0

