

Algoritmos

Kavanegasm

1. Desde el nodo 0

A.

- 1) 5
- 2) 3
- 3) 6
- 4) 8

B.

- 1) -
- 2) -
- 3) -
- 4) -

C.

- 1) 2
- 2) 1
- 3) 5
- 4) 7

2. Existe un camino con peso -4, que al pasar por los nodos 2 hacia el 1 tiene un costo de 2, del nodo 1 hacia el nodo3 tiene un costo de1 lo que suma un costo de recorrido de 4, al pasar del nodo 3 al nodo 2, el costo del recorrido llega a 0

Nodo	Costo total
0	0
2	3
1	5
3	6
2	2
1	4
3	5
2	1
1	3
3	4
2	0
1	2
3	3
2	-1
1	1
3	2
2	-2

1	0
---	---

3.

A.

Nodo/Iteración	0	1	2	3	4
0	0	0	0	0	0
1	INF	5	5	5	5
2	INF	3	3	3	3
3	INF	6	6	6	6
4	INF	8	8	8	8

B.

Nodo/Iteración	0	1	2	3	4
0	0	0	0	0	0
1	INF	5	4	3	2
2	INF	2	1	0	1
3	INF	6	5	4	3
4	INF	8	7	6	5

C.

Nodo/Iteración	0	1	2	3	4
0	0	0	0	0	0
1	INF	2	2	2	2
2	INF	1	1	1	1
3	INF	5	5	5	5
4	INF	7	7	7	7

4. El grafo b tiene un camino con costo -4 y por ende afecta encontrar el camino mas corto retornando como camino mas corto un peso negativo por lo tanto la función de Bellman Ford no calcula correctamente el camino mas corto y retorna en la última iteración en el nodo 2 un costo de 1 cuando en realidad es de -1

5. Para cada iteración en el for, todas las aristas (8), se llama 8 veces la función relax por cada iteración, son 4 iteraciones entonces en total se llama 32 veces dicha función. Si se utilizara la programación dinámica, almacenando el arreglo de los nodos en una tabla Hash de la función relax y así dejándolos por fuera de la siguiente iteración

6.

Nodo/Iteración	0	1	2
0	0	0	0
1	INF	5	5
2	INF	3	3
3	INF	6	6
4	INF	8	8

Nodo/dist	0	1	2	3	4
0	Min				
1	Min		Min	Min	
2	Min	Min	Min	Min	Min

Nodo/Iteración	0	1	2
0	0	0	0
1	INF	2	2
2	INF	1	1
3	INF	5	5
4	INF	7	7

Nodo/dist	0	1	2	3	4
0	Min	Min			
1	Min	Min		Min	Min
2	Min	Min	Min	Min	Min