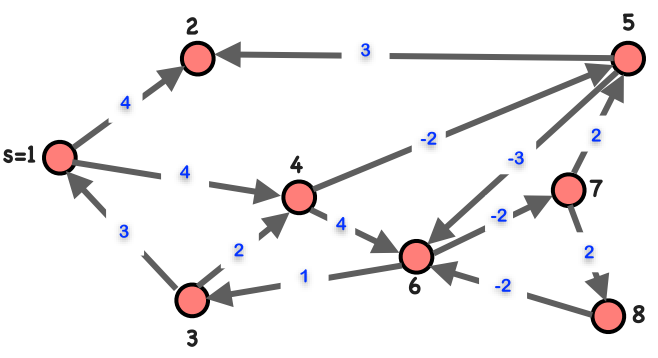
A. 完成投影片p30之Bellman-Ford演算法檢測負循環範例

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 0 | 0 | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ | ∞ |
| 1 | 0 | 4 | ∞ | 4 | ∞ | ∞ | ∞ | ∞ |
| 2 | 0 | 4 | ∞ | 4 | 2 | 8 | ∞ | ∞ |
| 3 | 0 | 4 | 9 | 4 | 2 | -1 | 6 | ∞ |
| 4 | 0 | 4 | 0 | 4 | 2 | -1 | -3 | 8 |
| 5 | 0 | 4 | 0 | 2 | -1 | -1 | -3 | -1 |
| 6 | 0 | 2 | 0 | 2 | -1 | -4 | -3 | -1 |
| 7 | 0 | 2 | -3 | 2 | -1 | -4 | -6 | -1 |
| 8 | 0 | 2 | -3 | -1 | -4 | -4 | -6 | -4 |

(p.s有變動只有前一次迭代前節點有變動者)  
  
B. 說明如何將匈牙利演算法能夠解答的最小成本指派問題(assignment problem)變轉為(reduce to)最大流問題  
加上source和target點，相連到二分圖的邊均為cost:0、capacity:1  
原先成本變為cost，capacity也為1，並做最小成本最大流運算，負成本者即為最小成本指派之項目

C. 將以下最小成本指派問題(assignment problem)轉換為流網(flow network)，以便使用最大流(max-flow)演算法解決之

|  |  |  |  |
| --- | --- | --- | --- |
|  | Task 1 | Task 2 | Task 3 |
| Carl | $24 | $28 | $24 |
| Bob | $26 | $32 | $28 |
| Alex | $24 | $28 | $30 |

D. 自行設計一個流網(flow network)(除s、t外具有4個節點，並具有10個edge)，解出其最小成本最大流

24

28

24

26

32

28

24

28

30

(1,24)

(1,28)

(1,24)

(1,26)

(1,32)

(1,28)

(1,24)

(1,28)

(1,30)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

轉換成最大流:加入soruce和target，連接edge皆為capacity:1、cost:0  
原先二分圖edge為cost，capacity皆為1

(1,24)

(1,28)

(1,24)

(1,26)

(1,32)

(1,28)

(1,24)

(1,28)

(1,30)

(1,-24)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

選擴增路徑，先滿足最大流，在符合最小成本(小於時才改變)

(1,24)

(1,28)

(1,24)

(1,26)

(1,32)

(1,28)

(1,24)

(1,28)

(1,30)

(1,-24)

(1,-24)

(1,-26)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

路徑2:cost 26-24+24 = 26

(1,24)

(1,28)

(1,-26)

(1,32)

(1,28)

(1,24)

(1,28)

(1,30)

(1,-24)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

(1,0)

24

28

24

26

32

28

24

28

30

(12,5)

(14,5)

(20,10)

(4,1)

(7,14)

(16,9)

(13,8)

(10,5)

(4,6)

(9,9)