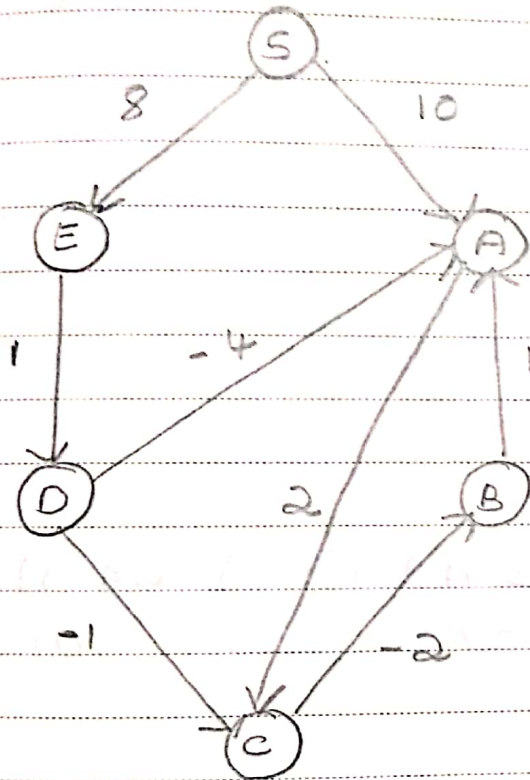


* Bellman Ford Algorithm *

Example 01: Case 01



write edges :

(S, E)

(S, A)

(E, D)

(D, A)

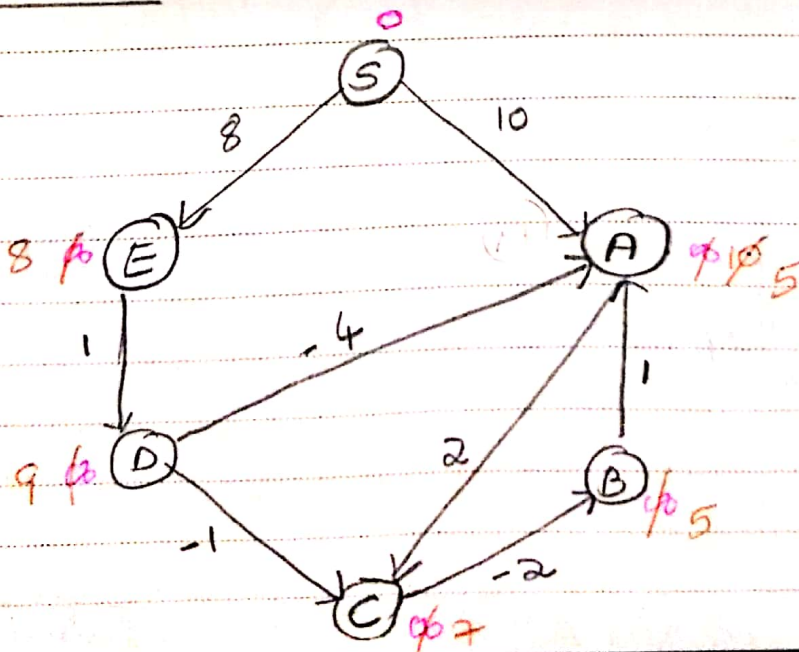
(A, C)

(B, A)

(D, C)

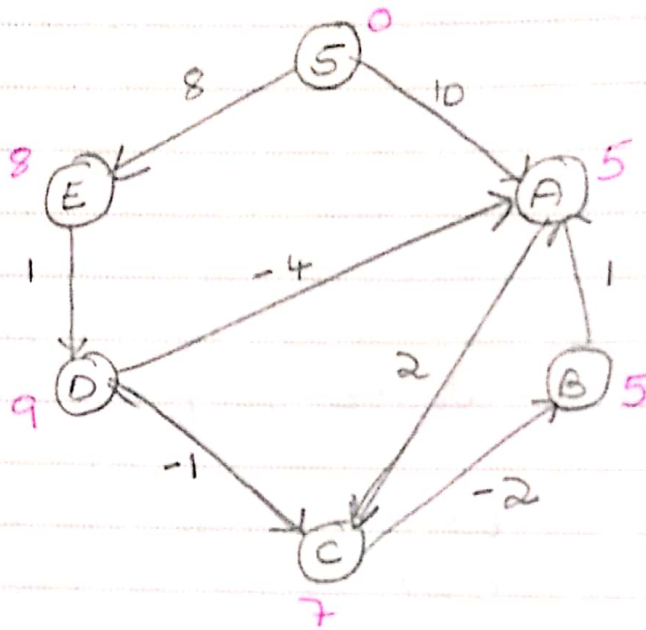
(C, B)

Iteration 01:



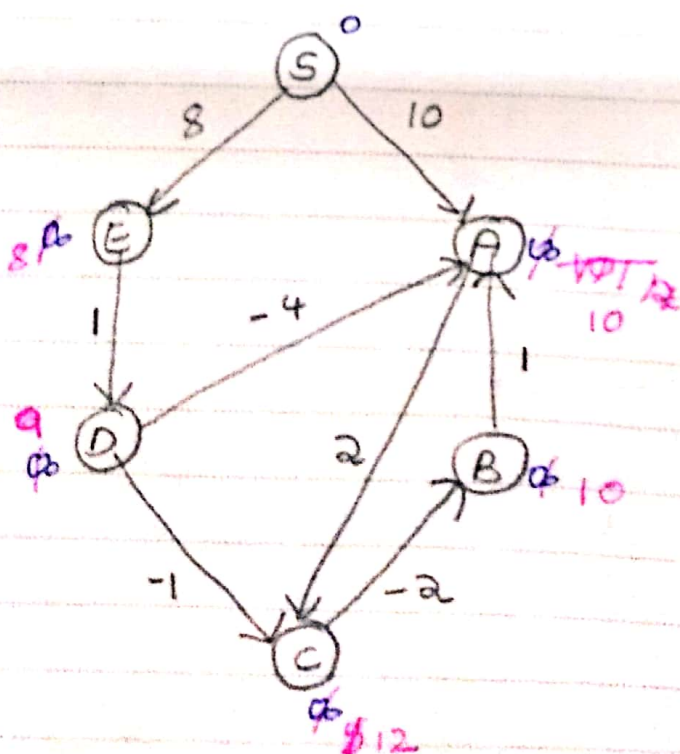
Iteration 02:

no changes, we stop.



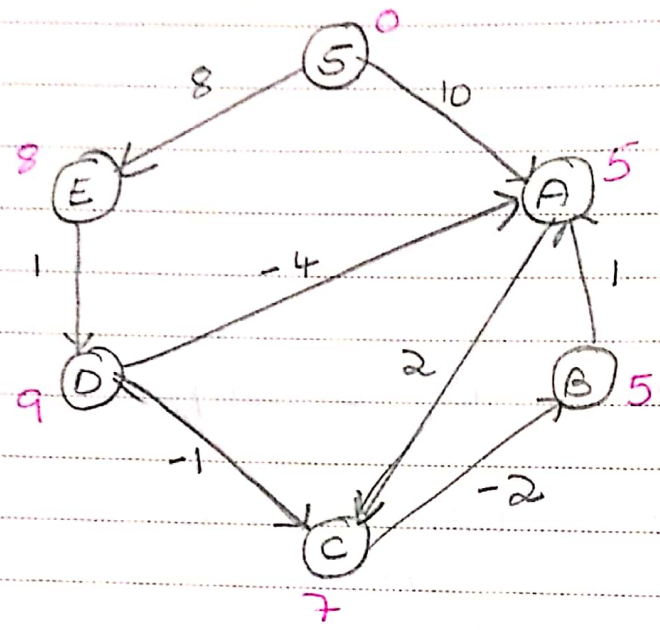
Case 02: Edge list: (S,E) (S,A) (A,C) (C,B), (C,A) (D,A) (D,C) (E,D)

Iteration 01:



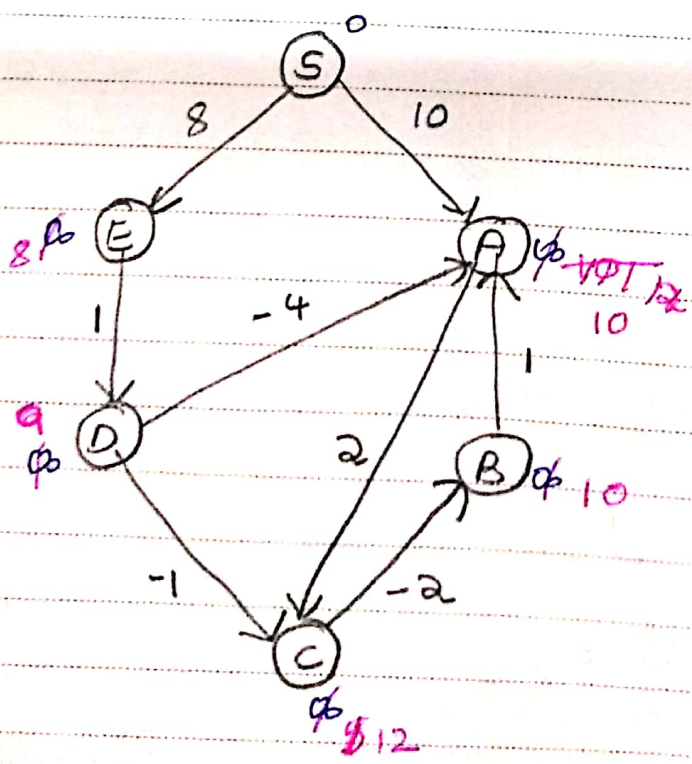
Iteration 02:

no changes, we stop.

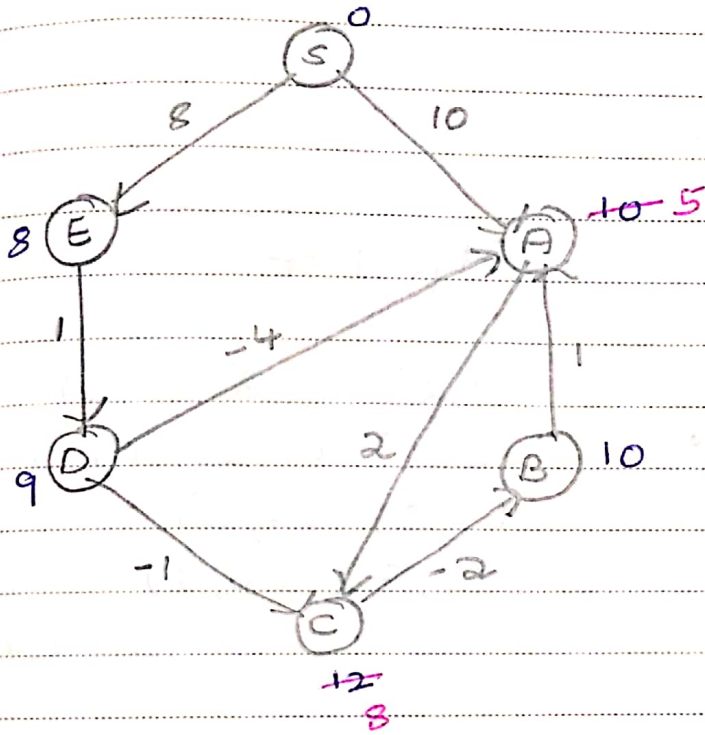


Case 02: Edge List: (S,E) (S,A) (A,C) (B,A)
(C,B) (D,A) (D,C) (E,B)

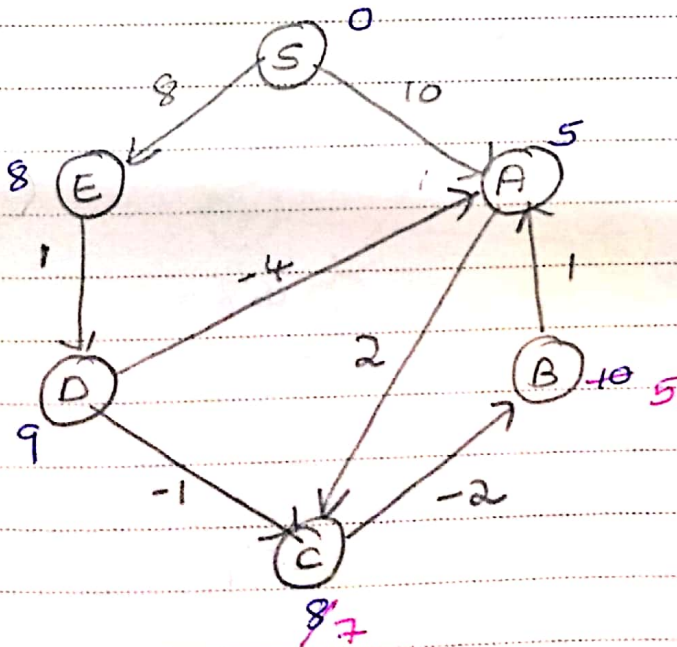
Iteration 01:



Iteration 02:



Iteration 03:

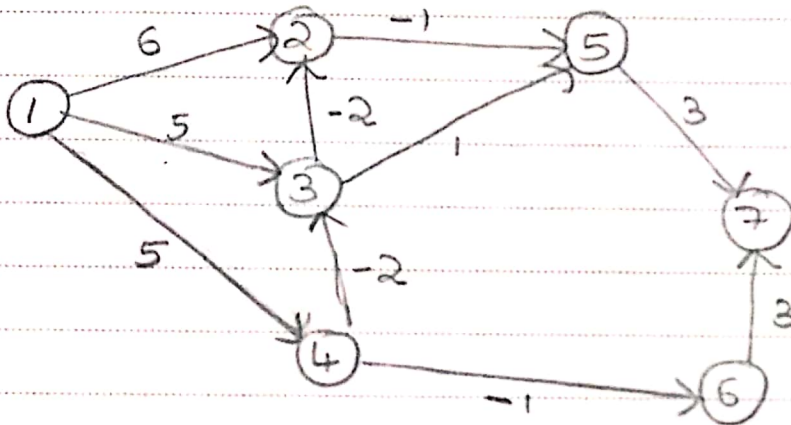


Iteration 04:

Nothing changes. We stop.

Note: With n vertices, we will have atmost $n-1$ iterations.

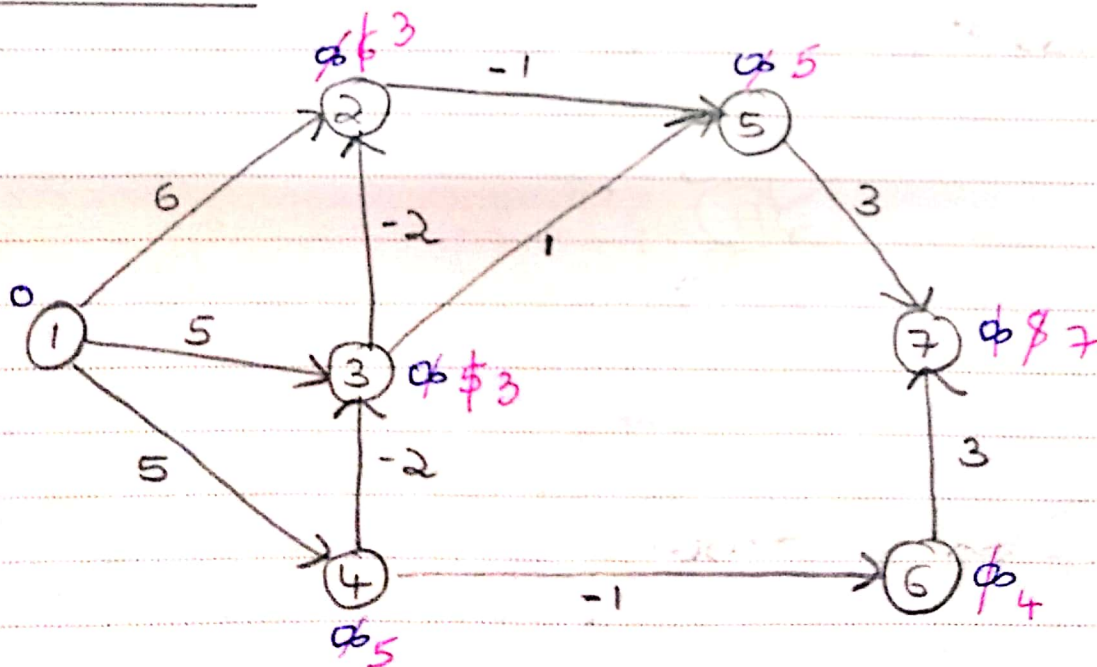
Example 02:

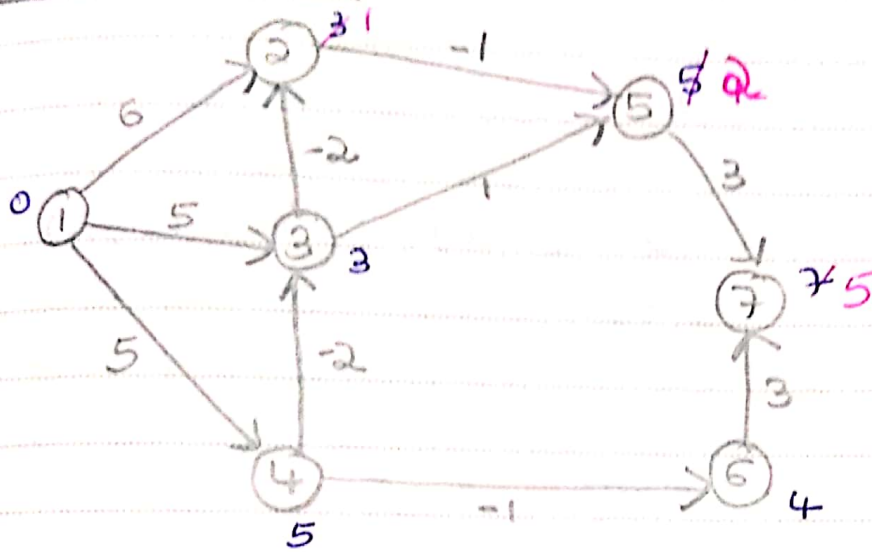
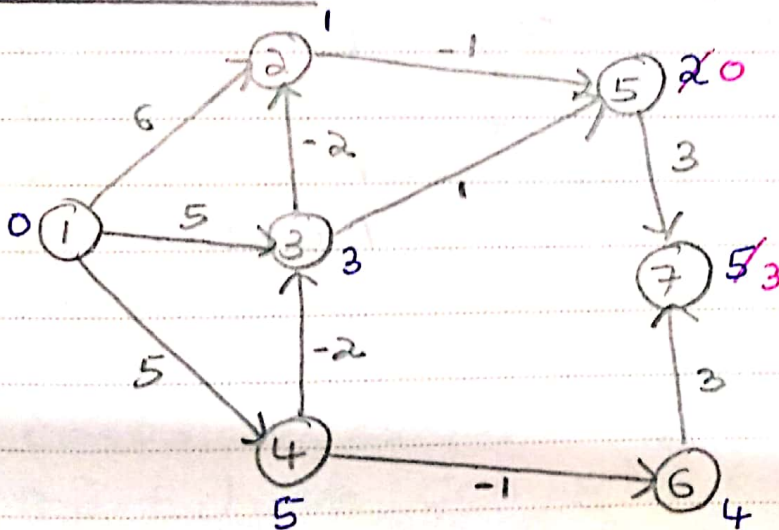
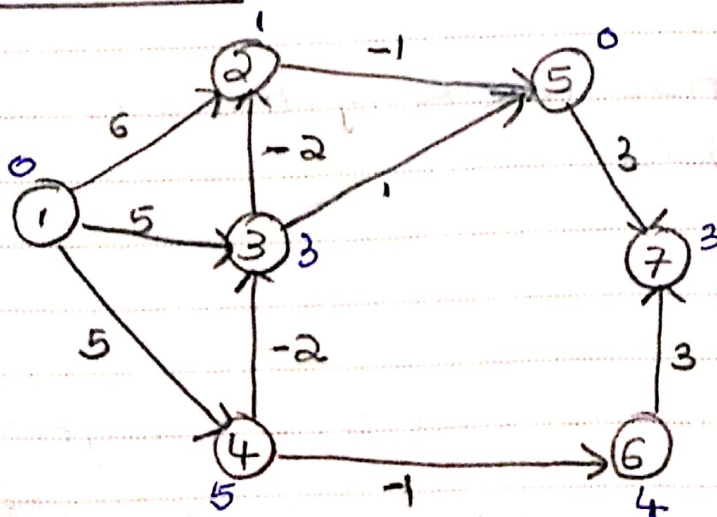


Edge list:

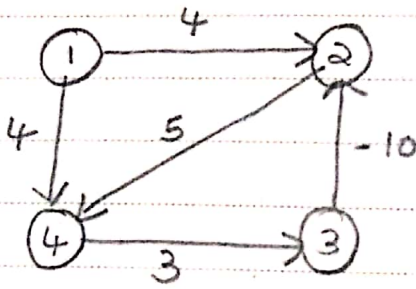
(1,2) (1,3) (1,4)
 (2,5) (3,2), (3,5)
 (4,3) (4,6)
 (5,7) (6,7)

Iteration 01:

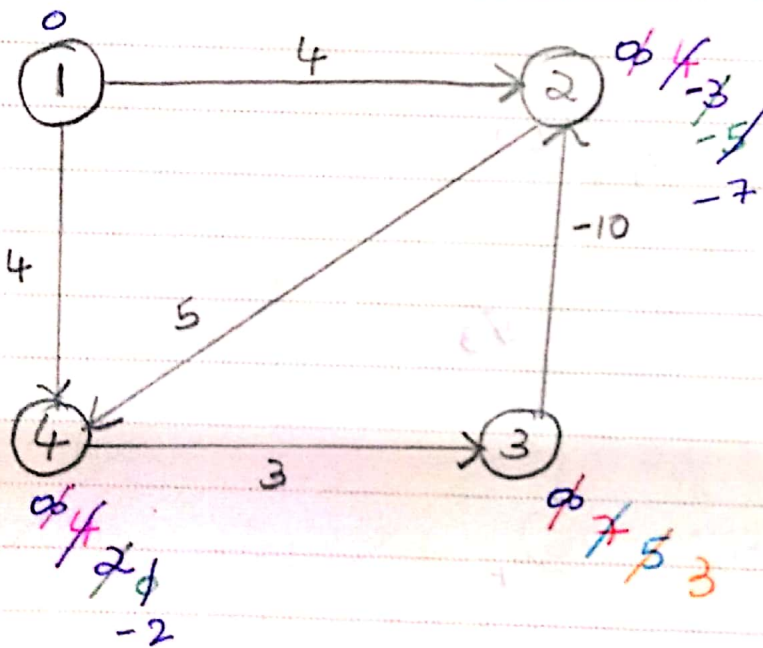


Iteration 02:Iteration 03:Iteration 04:

NO changes
we stop.

Example 03:Edge list:

(3,2) (4,3) (1,4) (1,2)
(2,4)

Iterations

Iteration 01

Iteration 02

Iteration 03

Iteration 04

Iteration 05

Iteration 06

Iteration 07

&

So on.

Bellman-ford will not work if there is a negative weight cycle.

In the graph:

4 → 3 → 2 → 5 → 4 is a negative weight cycle.