

PROJECT ON

IMPLEMENTATION OF SOME VARIANT OF THE BELLMAN-FORD (ROUTING) ALGORITHM

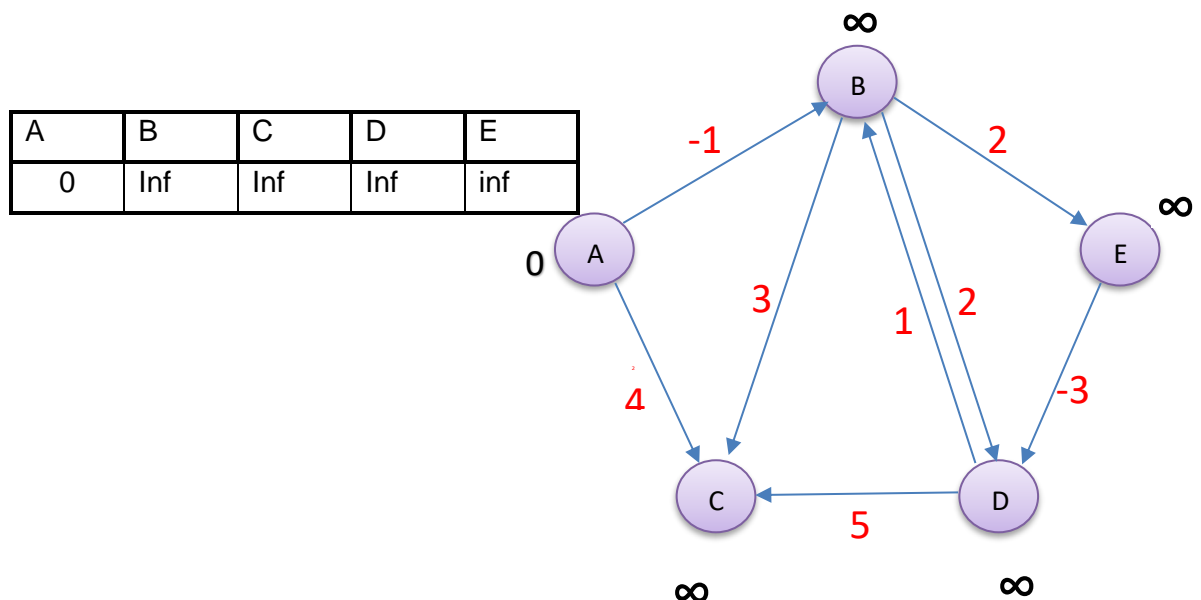
By: Sakshi Agarwal
: Ajeet Kumar
Guided by : Dr. Siddhartha Sarma

Course Name: Programming
Practicum
Course Code: CS571

STEPS for finding the shortest distance to all vertices from the source using the Bellman-Ford algorithm using an example:

Let's assume that A is a source and E is a destination, So we are trying to find the shortest distance between the source (A) to destination (E)

- 1) Let the given source vertex be 0. Initialize all distances as infinite, except the distance to the source itself. The total number of vertices in the graph is 5, so all edges must be processed 4 times.



Cost = cost(source) + Weighted edge

If (cost(source) + Weighted edge < cost(destination)):

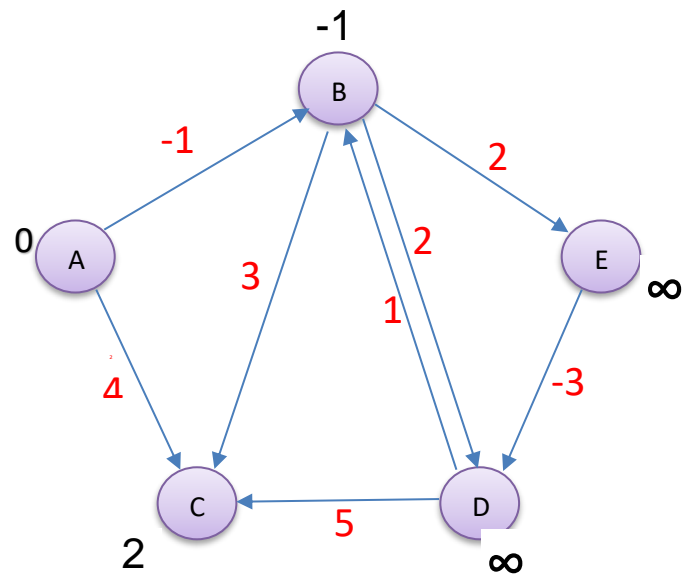
Hence, we will assign this Cost to vertex B.

else:

The cost of vertex B remains the same cost.

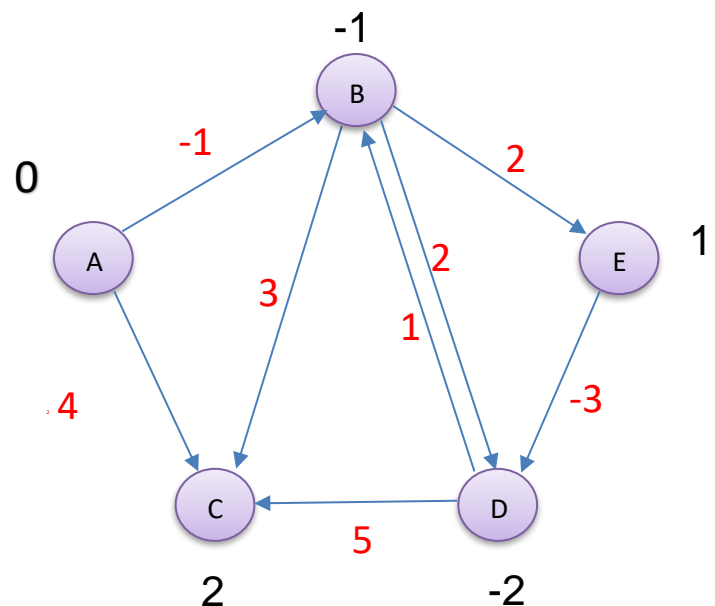
- 2) Let all edges be processed in the following order: (B, E), (D, B), (B, D), (A, B), (A, C), (D, C), (B, C), (E, D). We get the following distances when all edges are processed the first time. The first row shows the initial distances. The second row shows distances when edges (B, E), (D, B), (B, D), and (A, B) are processed. The third row shows distances when (A, C) is processed. The fourth row shows when (D, C), (B, C), and (E, D) are processed.

A	B	C	D	E
0	Inf	Inf	Inf	inf
0	-1	Inf	Inf	Inf
0	-1	4	Inf	Inf
0	-1	2	Inf	inf



- 3) The first iteration guarantees to give all shortest paths which are at most 1 edge long. We get the following distances when all edges are processed a second time (The last row shows the final values).

A	B	C	D	E
0	Inf	Inf	Inf	inf
0	-1	Inf	Inf	Inf
0	-1	4	Inf	Inf
0	-1	2	Inf	inf
0	-1	2	Inf	1
0	-1	2	1	1
0	-1	2	-2	1



- 4) The second iteration guarantees to give all shortest paths which are at most 2 edges long. The algorithm processes all edges 2 more times. The distances are minimized after the second iteration, so the third and fourth iterations don't update the distances.

GitHub Link :- <https://github.com/Sakshiagarwaliitm/The-Bellman-Ford-Algorithm->