* **Problem** : <https://ideone.com/7dJhIl>
* **Approach** : **Bellman Ford finds shortest path to all Vertices REACHABLE from SRC vertex.(Can detect neg weight cycle)**

(If it’s not reachable ,distance remains INF).

**-> HERE WE NEGATE ALL THE EDGE WEIGHTS AS WE WANT SINGLE SRC LONGEST PATH.**

**(IMP)(Can do the same using dijkstra too,as the weights were originally positive,but Dijkstra will still fail when there are Neg- Weight Cycles).**

* **Algo :**

**-> Step - 1 Relax all Edges (V-1) times.And you get shortest distance to all nodes.**

-> Some Intuition before step 2 : If on relaxing all one more time i.e Vth time, one of the node’s distance still decrease means there is definitely a Negative Weight Cycle(which includes a node which is reachable from src and that node is reachable to destination).

-> And so we can get as short path as we want,as distance decrease on every round.

**-> Step 2 : (Imp) So algo says after step 1 ,relax all edges (V-1) times AGAIN ,and if neg weight cycle detected through a node, Update that node’s distance to -INF ,**

**So now after (V-1) rounds complete ,distance of all nodes (IF REACHABLE FROM THAT NEG WEIGHT CYCLE) including dest. will be updated to -INF.**

Thus if src is reachable to neg weight cycle and neg weight cycle is reachable to destination,then at end dest node dist. will definitely be -INF,and so we can print (-1) as demanded in the question.

**So total ‘E’ edges relaxed ‘2V-2’ times ,so Time = O(V\*E).**

In worst case when E=O(V^2) then Time=O(V^3)

* **CODE :** <https://ideone.com/7dJhIl>