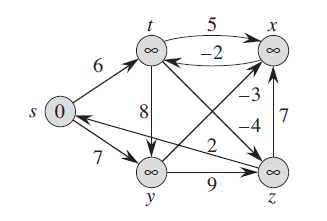
**Name: Khushi Nitinkumar Patel**

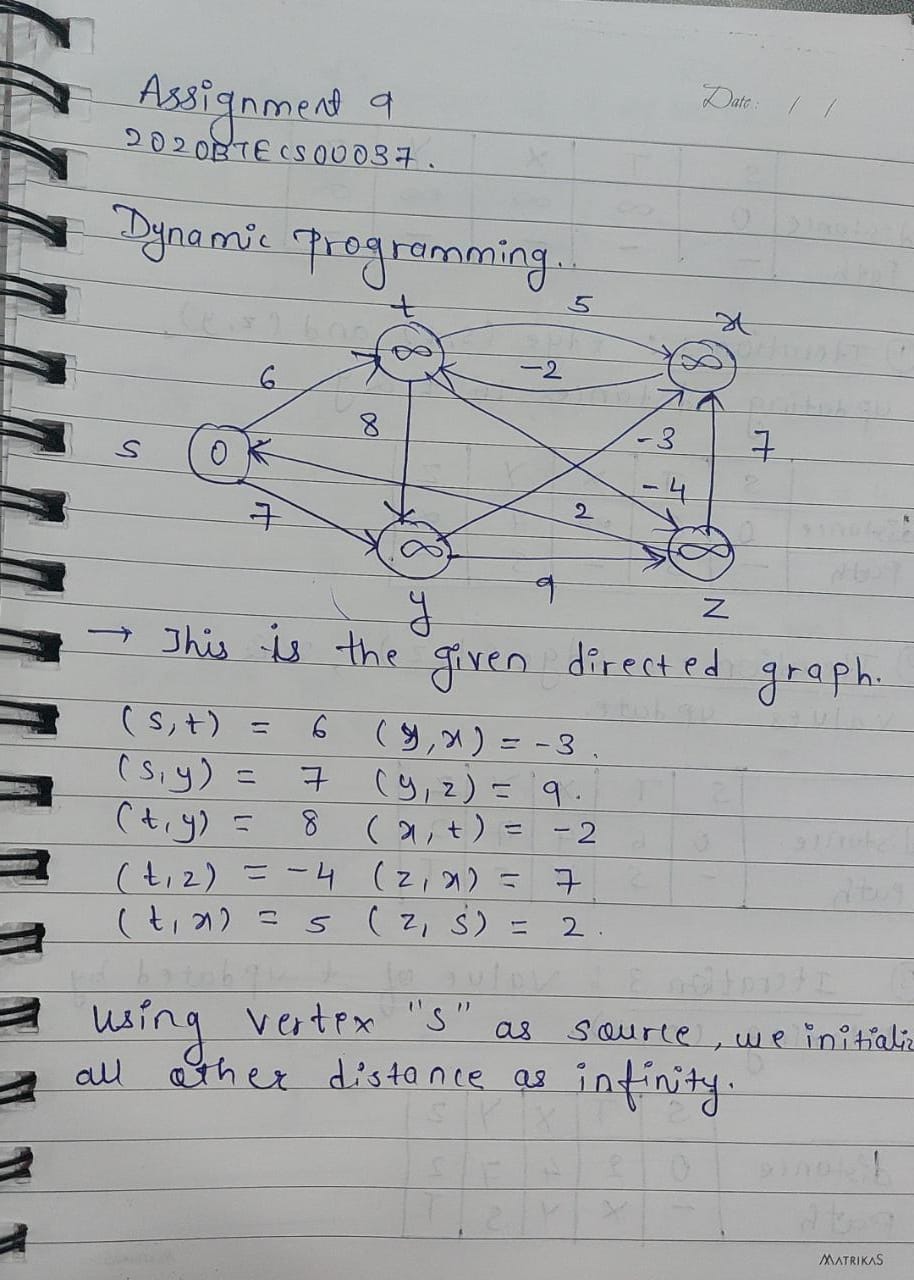
**PRN: 2020BTECS00037**

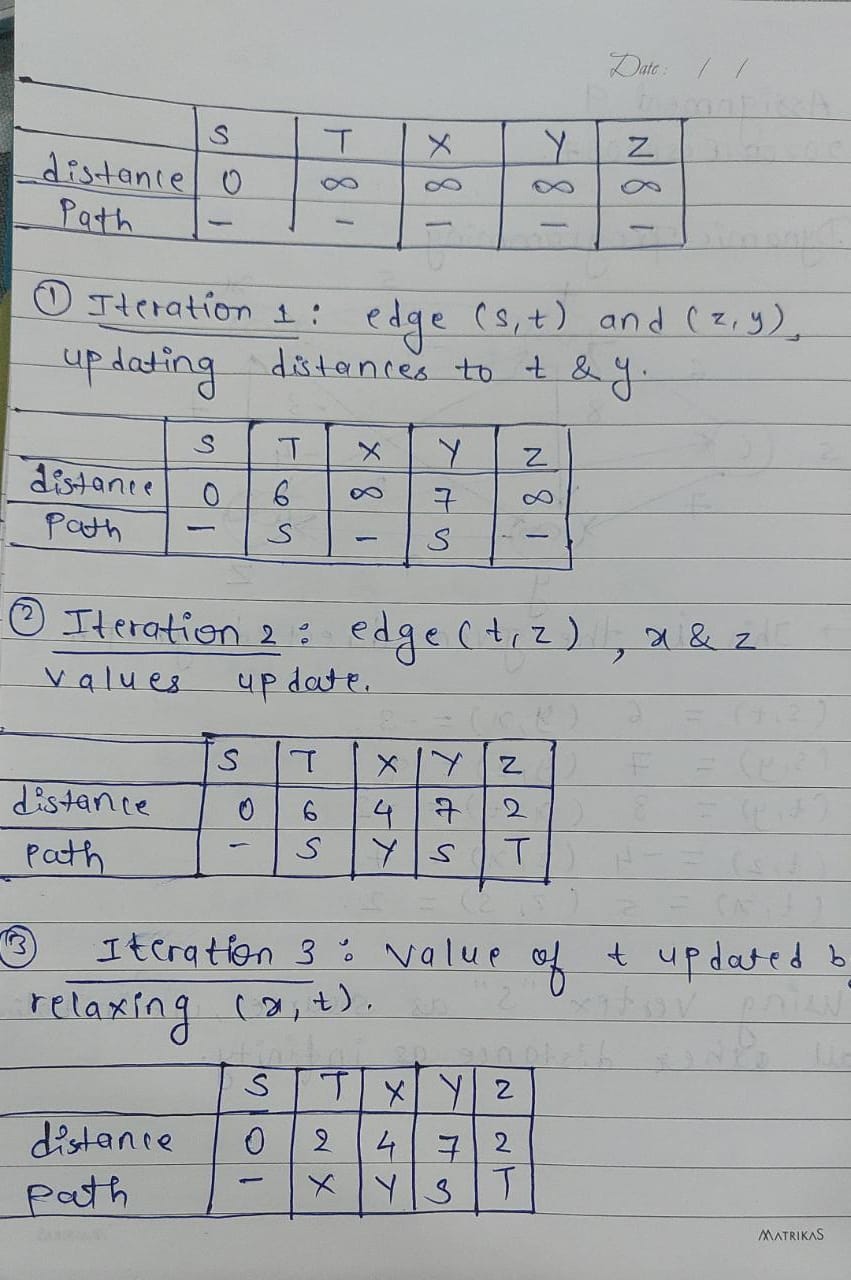
**Batch: T5**

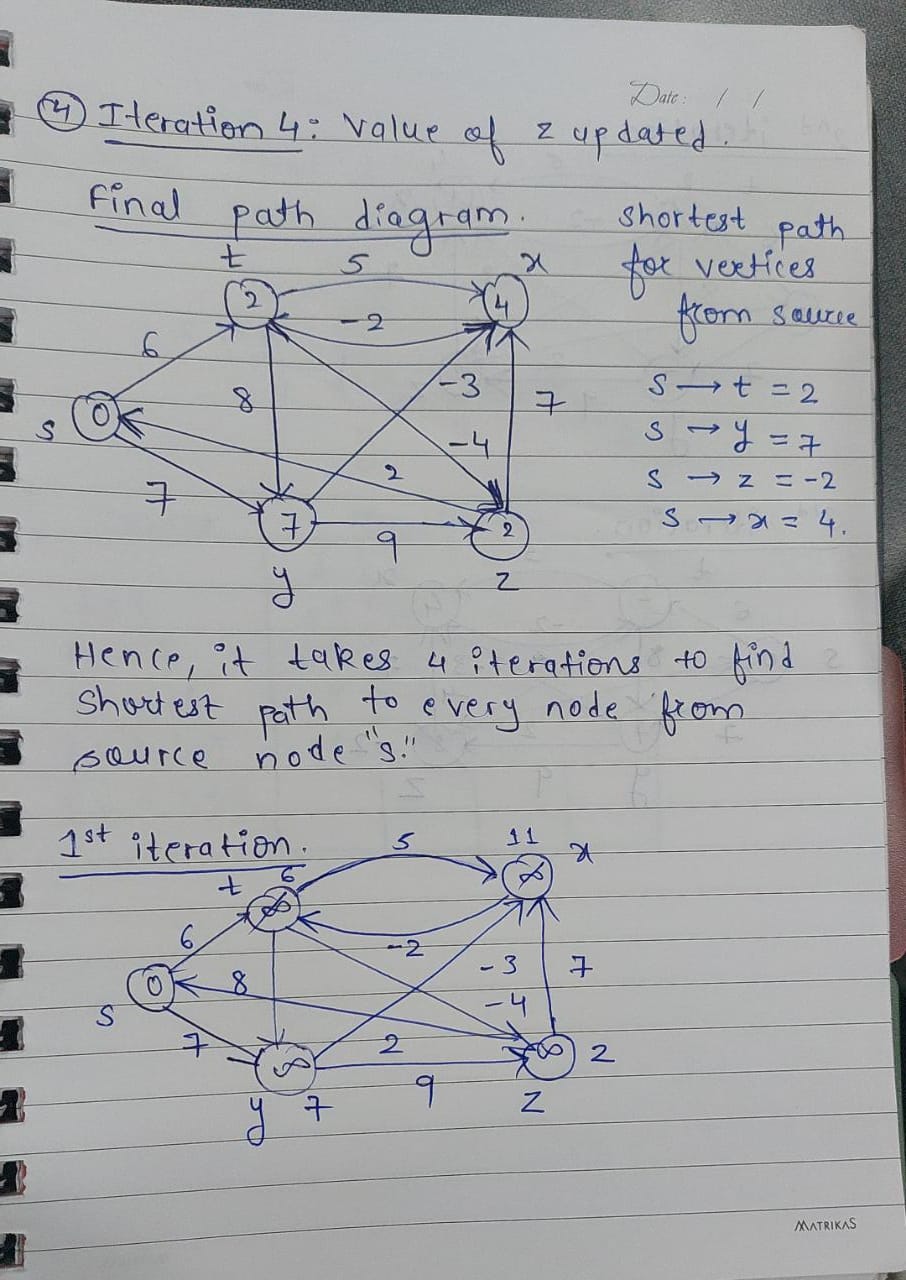
**Assignment no 9: Dynamic programming.**

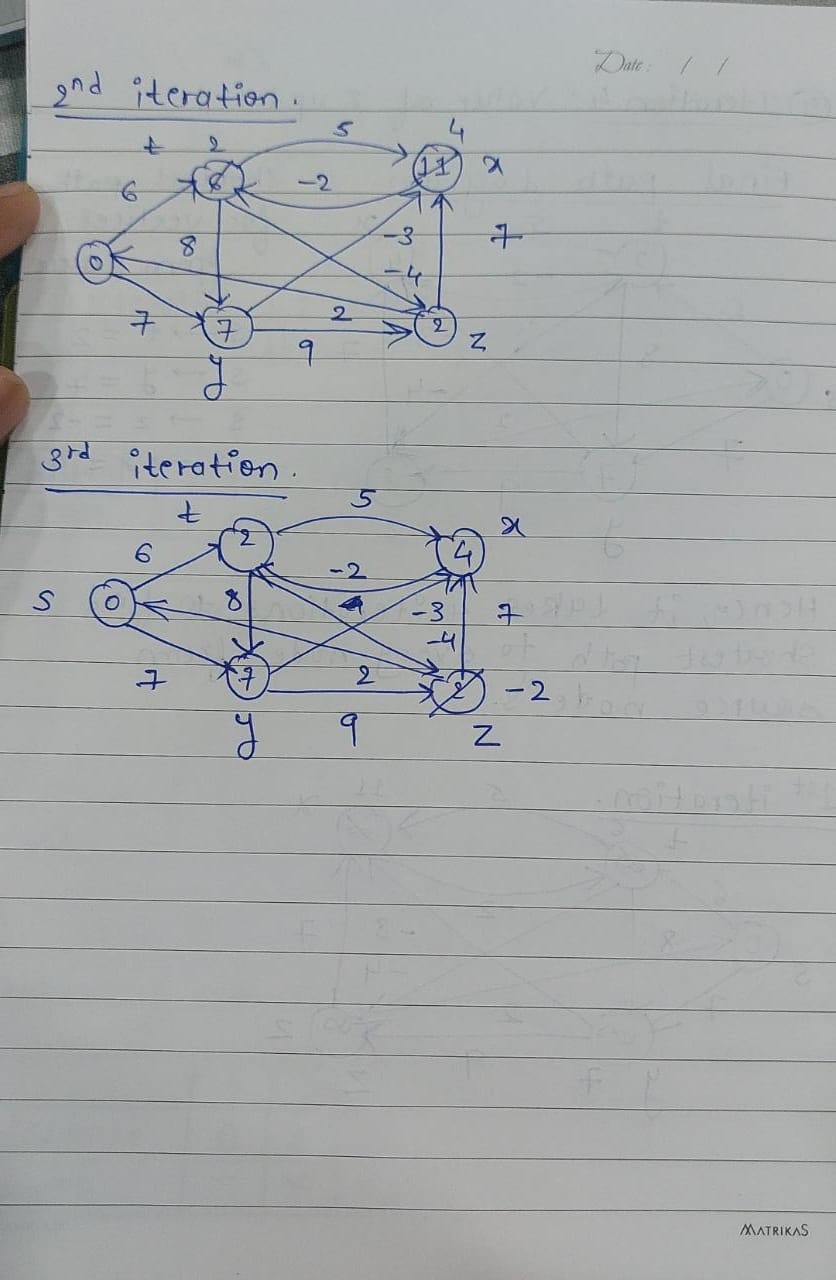
1. From a given vertex in a weighted connected graph, Implement shortest path finding Bellman-Ford algorithm.



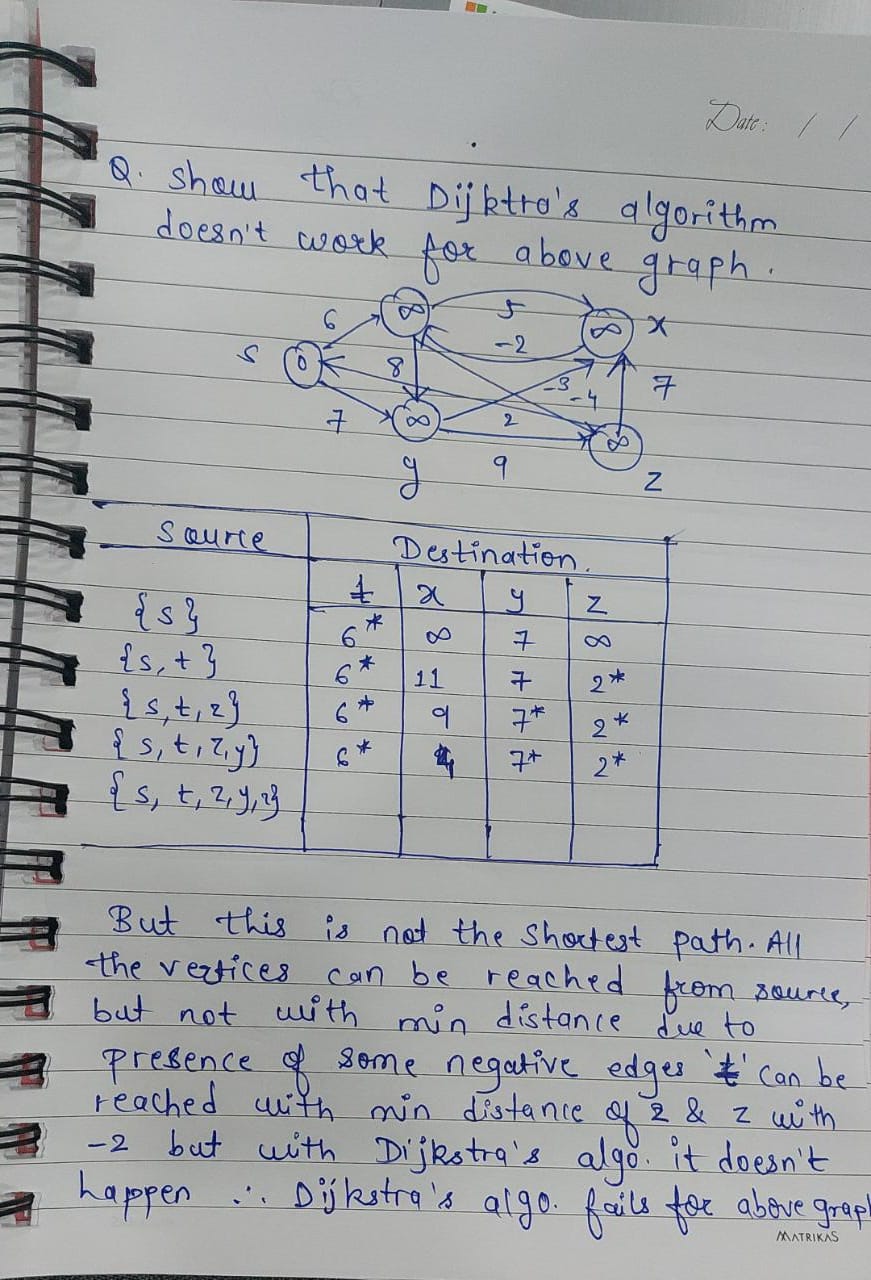


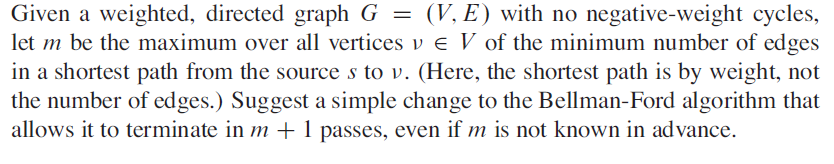
****

****

****

Q) Show that Dijkstra’s algorithm doesn’t work for above graph

****

Q) 

Path relaxation property of Bellman-ford implies that every vertex in the graph has achieved shortest path weight in “v.d” after m-iterations. But we don’t know for sure that no d value will change in (m+1)th iteration so we cannot terminate it at min iteration. So we can make a Bellman-ford algorithm such that it will stop when nothing changes after (m+1)th iteration.

The change to the Bellman-Ford algorithm to implement this optimization is:

Check if v was relaxed or not.

If v is relaxed then we wait to see if v was updated (which means being relaxed again).

If v was not updated, then we would stop