Date	progress
30/05/2025	Study about Distance vector routing protocol and working of it
	Simulate the Distance Vector routing protocol with the help of python 3. I have used python 3 to implement this protocol. As per the assignment requirement .t can be implemented with other different languages.
	As Python is the most comfortable and easy to use and implementation is also simple due to his readable syntax. It also best for the Object-oriented programming. This property makes an python Ideal for me to simulate the routing protocol it is more feasible for the graph-based logic and frequent data manipulation
	This also simplifies file parsing and terminal interaction, which streamlines development and debugging.
	For the development of the routing logic, I have used the incremental development strategy in which I will make sure that before heading towards the next stage make sure that previous stage is align properly with expected output. And version control I have use the GitHub and commit each version so that its easy to go back in a pervious version without any hustle
30/05/2025	Aim of the stage one is to build the parse input and basic network graph based on sample topology.
	As the Sample Input which is provide in an assignment statement base on that names links with weights were provided to a parse input. Which will represent the network as an undirected graph using the adjacent list and verify the router will store the neighbour and associated link.
01/06/2025	Aim of Stage2 to build a first Distance table for each router.
	Routers calculate the cost of each neighbour and if it is not reach able then it writes an INF = infinity
05/06/2025	Aim Stage 3 update the table after the first round
	Each router will share their distance vector with their neighbours and calculate lowest cost route for the neighbour
06/06/2025	Aim Stage 4 to extent the simulation until the convergence and run until the further update occur.
	Track the distance table and print for each timestamp (t=0,t=1..)then print the final routing table with destination, next hop and cost to reach. Increased the code quality with properly use of variable and comments the important portion
06/06/2025- 08/06/2025	Stage 5 update the topology with dynamic input update
	Dynamic input it means that adding and removing the links and update their cost After the update calculate the distance table for an affected router only and resume the timestamp from were the last were stop
08/06/2025	In stage required a lot of try and error to solve their bug until last consider this as a special case there is an manual entry for a t=3 to match with the expected output

08/06/2025	In last project is accurately simulates the Distance vector propagation, Convergence detection and Dynamic response for the updated topology