

## 6 Future Works

---

We have computed the longest depth between input node and output node using DFS algorithm. The time complexity may decrease if we use dynamic programming techniques with Dfs algorithm. But it requires extra space of  $O(n^2)$  when implementation is done using adjacency matrix. We tried to implement dynamic programming techniques using adjacency matrix representation of graph, we observe that there is a reduction in time complexity for small graphs. But when we use this technique for very big graphs like 30,000 nodes graph, this became infeasible because of memory requirements. So, implementation of dynamic programming may work fine if it is used with adjacency list representation of the graph. One other modification can be suggested as if Dijkstra algorithm is used instead of DFS algorithm then it may give better results for dense graph, but it also increases the space complexity and can't be used for a graph with edge cycle. For sparse graph, Dijkstra algorithm may not give a better result than DFS algorithm.