CSCI - 665 Foundations of Algorithms HW - 06

Authors - Sagar Kukreja, Rohan Shiroor

Problem 1:

Algorithm Used : Bellman Ford Algorithm

Proof of correctness :

The Bellman Ford Algorithm finds the shortest path to every vertex in the graph. In this modified Bellman Ford I am trying to control the iterations so that there is a single update for each path. This adds only one edge for every shortest path. I run the algorithm for k+1 iterations where k being the number of veils. I also keep a duplicate list and make updates only to the duplicate list so that there never 2 updates to a single shortest path. This finds the most optimized shortest path upto k+1 iterations which is the problem.

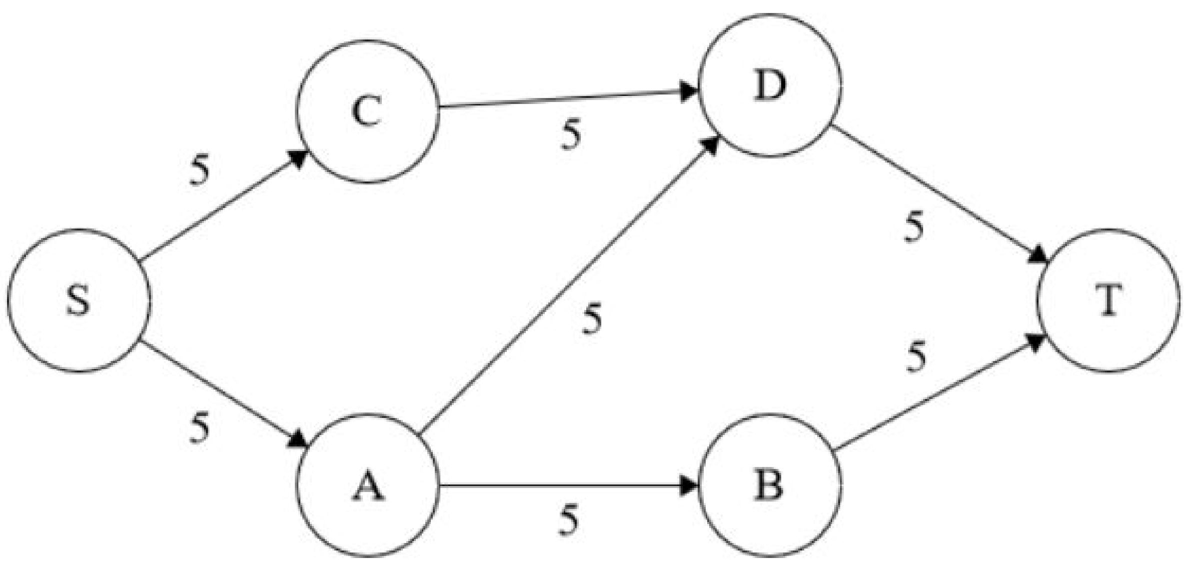
Time Complexity :

1) Reading in the input: O(m) + O(1) = O(m)
2) Creating a new graph i.e undirected graph from directed graph O(2m)
3) Initializing the array: O(n)
4) Updating the shortest path: O(nk)+O(2mk) = O(mk)
5) Validation: O(1)

Total: O(m) + O(2m) + O(n) + O(mk) + O(1) = O(mk).

Problem 2 :

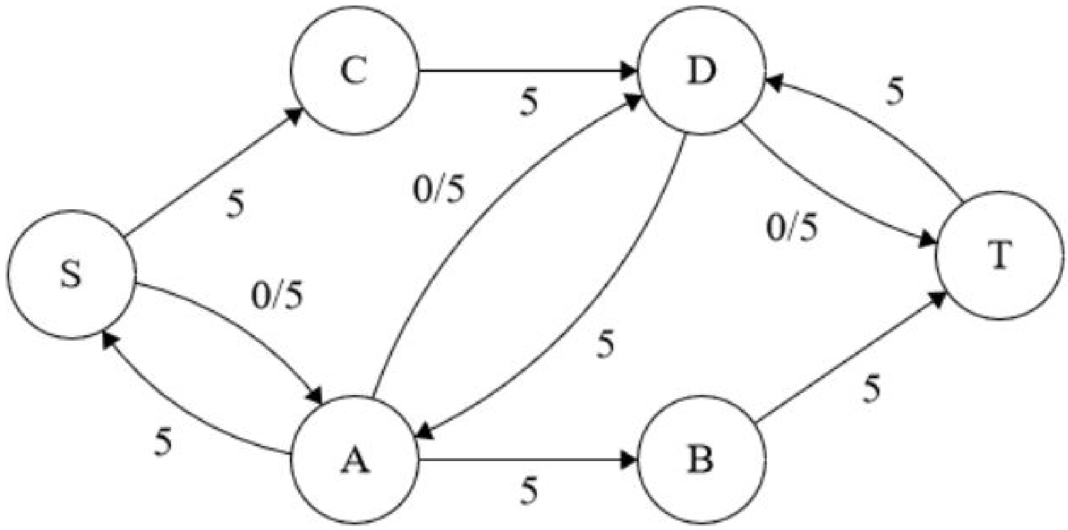
Original Graph



This is the flow graph.

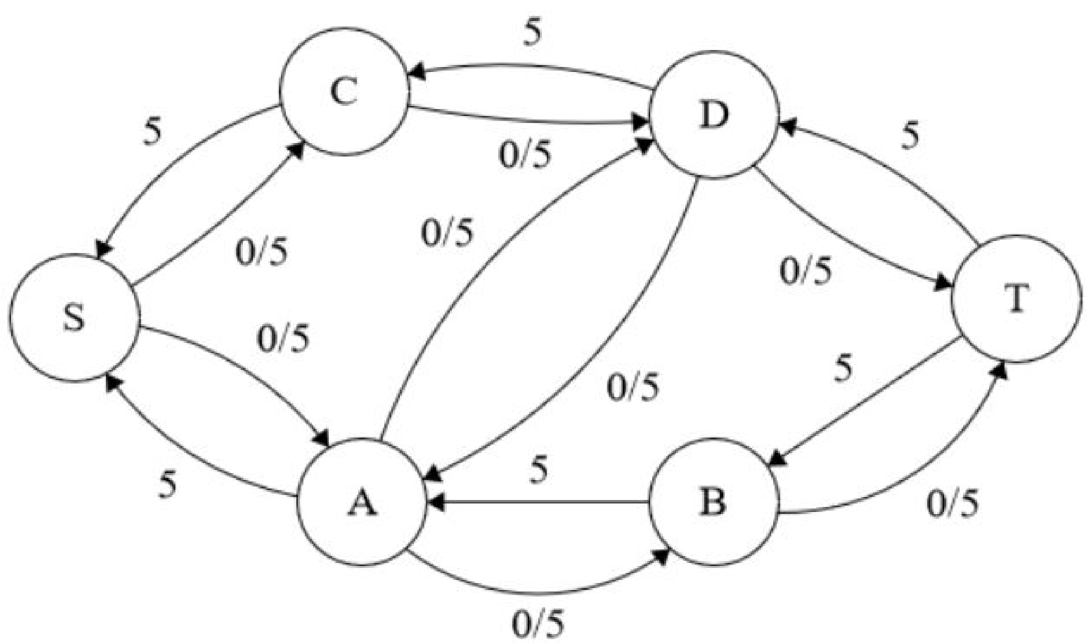
Iteration 1:

The residual graph after the 1st iteration is given below.



The current max flow after the 1st iteration is 5.

Iteration 2:



The max flow after the 2nd iteration is 10 after the 2nd iteration.
As seen from the graph the Edmond Karp algorithm uses a backward edge to find the maximum flow in the graph.

Problem 3:

Proof of Correctness:
This is a problem of finding max-flow through multiple source and multiple sink in Graph. This can be converted to single source and single sink by adding extra node and connecting it to all students with flow of 3 units and connecting all courses to one sink with their given capacities. Now we can apply ford-fulkerson algorithm to find the maximum flow through graph.

Adding a source which connects all student nodes and adding a sink that connects all courses makes it a single source- single sink problem. The flow to all student nodes are made 3 which make sures that no student gets more than 3 subjects. And the value of course intakes are assigned to flow from courses to sink which makes sure that every course enrolls only those many as its capacity.

Complexity:-

1) We use BFS to find path as it always picks a path with minimum number of edges.- $O(v^2)$ as we use adjacency matrix. This version of Ford-fulkerson where BFS is used is Edmond-karp Algorithm.
2) Ford-Fulkerson is then applied on each path with updated residual graphs. Complexity for one iteration - $O(EV)$

Total Time complexity - $O(EV^3)$

Problem 4:

a) Given G(V,E) with |V| = n, we can define cost(edge) = 0 for all $edge \, \varepsilon \, E$. Then we add edges E' to G to make it complete graph and assign cost(edge) = 1 for all $edge \, \varepsilon \, E'$. We can do this in polynomial time.

b) Now, we pass this graph to TSP black box. If the answer to TSP cycle of cost at most 0 is yes then, we know there is a cycle that visits all nodes exactly once. The edges used are from E, hence we can say that there is an Hamiltonian Cycle in G.

   If the cost is > 0 , we know it used edges from E'(as new edges have cost of 1 each) and hence it doesn't have Hamiltonian Cycle using original edges.