9101 Assignment 4
Haojin Guo
z5216214

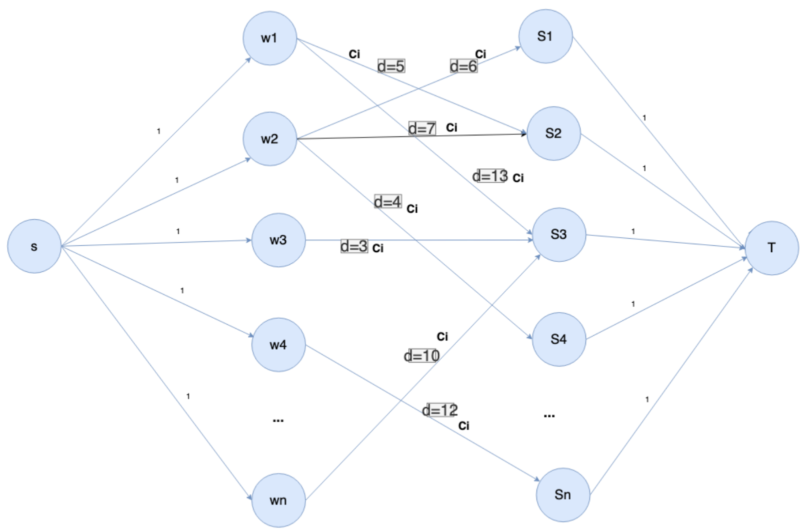
Q2.

2. You have n warehouses and n shops. At each warehouse, a truck is loaded with enough goods to supply one shop. There are m roads, each going from a warehouse to a shop, and driving along the ith road takes d_i hours, where d_i is an integer. Design a polynomial time algorithm to send the trucks to the shops, minimising the time until all shops are supplied. (25 pts)

Hint: Combine a binary search with a max flow. By sorting you can assume that d_i form an increasing sequence. Fix i and consider only roads which take $\leq d_i$ hours to travel from a warehouse to the corresponding shop and use max flow to see if they are enough to obtain a matching of warehouses with shops which is of size n. Use a binary search on i to find the smallest d_i which meets the requirements.

Solution,

1) Merge sorting the m roads in ascending order based on distribute d_i.
2) Construct a flow network as a directed graph and it should be a bipartite graph.



a) All the warehouses as vertices on the left: $W_i - warehouses$
b) All the shops as vertices on the right: $S_i - shops$
c) Each road between a warehouse and a shop is represented by an edge.
d) Setting a super source S and a super sink T, and their capacity is 1.

3) Performing binary search on the sorted list on 1) about d_i, suppose low = road with smallest d_i, high = road with highest d_i
   a) Let $mid = d_i = (low + high) // 2$.
   b) For all the road, if a road (edge) of d_j between a warehouse and a shop is smaller or equal $mid (d_j \leq d_i)$, then the capacity of this edge is set to 1, otherwise, the capacity of this edge is set to 0.

c) Compute the maximum flow (=maximum cut) by the extension of Preflow-Push algorithm.

i. If the maximum flow = n, which the current selection of d_i satisfies the requirements of the question, and then it is need to continue the binary search, that is let high = mid, and repeat the above operations a) b) c).

ii. Otherwise, let low=mid, and repeat the above operations a) b) c).

iii. Until we find the smallest value of d_i that meets the requirements of minimizing the time from warehouse to shops.

4) Time complexity.

a) Merge sorting: $O(mlogm)$

b) Binary search: $O(logm)$

i. Max-flow algorithm: Preflow-Push algorithm: $O(|2n+2|^3) = O(n^3)$

Therefore, the total cost is $O(mlogm) + O(n^3)$