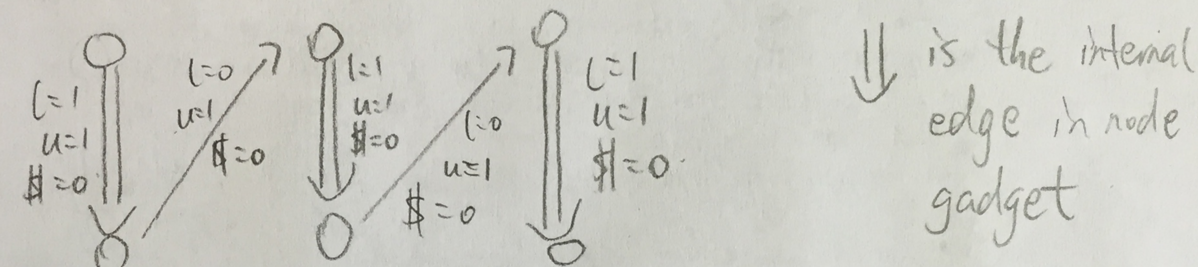
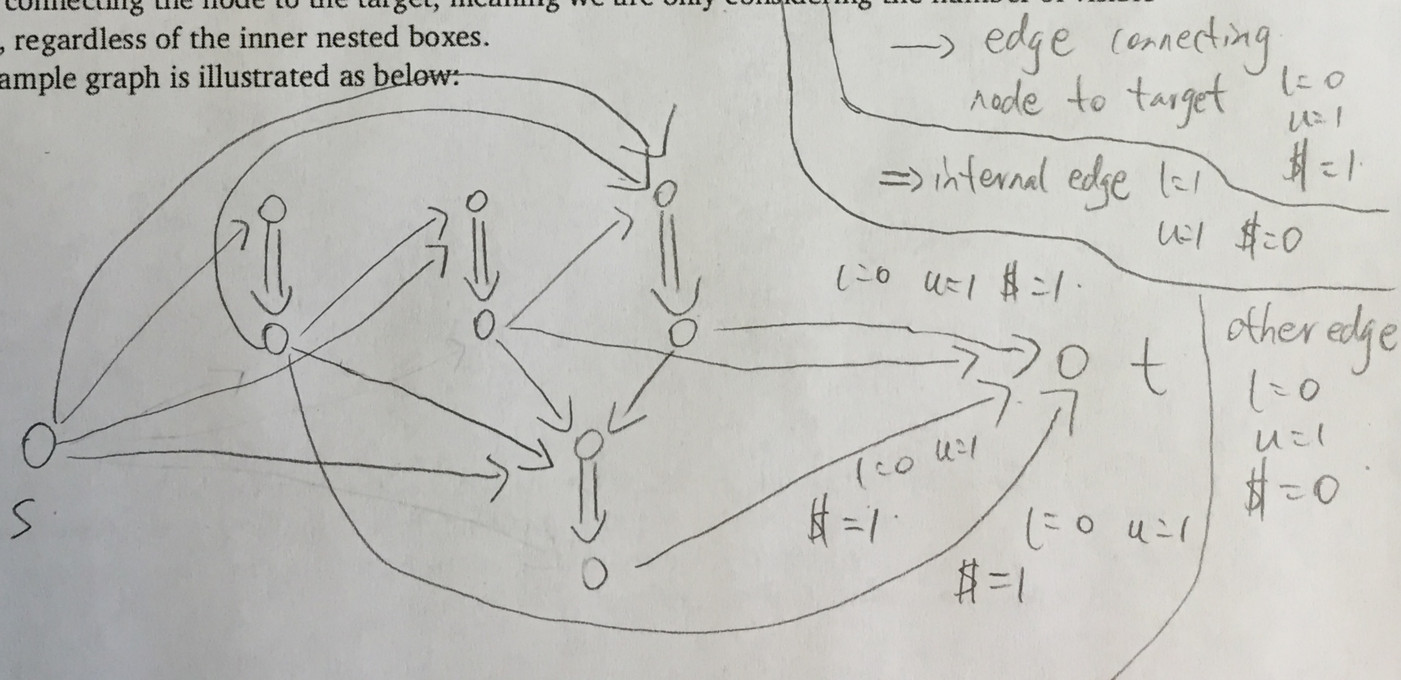
to Part A): We represent each box as a node in the graph, and draw a directed edge from u to v if box u can be put into box v. Then put a directed edge from source to every node and a directed edge from every node to the target. Note that if there is flow in the edge connecting the node to the target, it means that the box that the node represents is visible. Set the capacity of all edges to be $l(e) = 0$ and $u(e) = 1$.
Then the 1st requirement can be met by introducing the following gadget on every node. The idea is to expand each node into two sub-nodes, one sub-node for the incoming edges and one sub-node for the outgoing edge. Then connect the incoming sub-node to the outgoing sub-node and put the lower bound $l(e)$ and upper bound $u(e)$ of capacity of the connecting edge both to be 1. This guarantees that the flow through each node (representing of box) is 1, meaning that each box is only nested only once. An illustration is as below :



The second requirement can be met by setting the cost of edges that connect the outgoing sub-node of every node to be 1 and setting the cost of all other edges to be 0. This guarantees that no matter how the boxes are nested inside each other, the cost will only be affected by the edges connecting the node to the target, meaning we are only considering the number of visible boxes, regardless of the inner nested boxes.
An example graph is illustrated as below:



This method works because by introducing the sub-node for each node and by setting the $l(e)$ and $u(e)$ of the inner edge to be 1, we ensure that the flow through each node is 1 meaning that each box is either nested into other box or taken as a visible box. And since only the cost of the edges connecting the node to the target is 1 (if there is flow in the edge connecting the node to the target, it means that the box is visible), by solving for the min cost flow problem, we will know what is the minimum number of visible boxes.