Homework 9. CS430.

Junzhe Zhang.
A20254389.

1. Problem 1. Problem 22.4-2 CLRS

First, we do topological sort on the graph G. using DFS.

~~Suppose~~ Suppose v is any node in graph G. and v has k incoming edge, $P_1, P_2 \cdots P_k$ from node $u_1, u_2, u_3 \cdots u_k$ respectively.

thus, there are $P_1 + P_2 + P_3 \cdots + P_k$ path from s to v.

From first step, we maintain a list $v_1, v_2 \cdots v_n$ such that.

v_i represent the number of edges into node i from node j, $j < i$.

Then processing node by the list order. when v_i, if $v_i = s$. then $P[v_i] = 1$. if $v_i \neq s$, $P[v_i] = \sum_{v_j : [v_j, v_i] \in E} P[v_j]$, this represents that we sum up all $p[v_j]$ ~~values~~ values if for a node v_j, exists an edge (v_j, v_i).

Last. we return $P[t]$.

Time complexity. Topological $\Theta(E+V)$. then we run over all nodes. For each node, we check for all possible incoming edge, thus. $O(E+V)$

$$\Theta(E+V) + O(E+V) = O(E+V).$$