

## Module 5–6

Name: NAME HERE Collaborator: COLLABORATOR NAME HERE

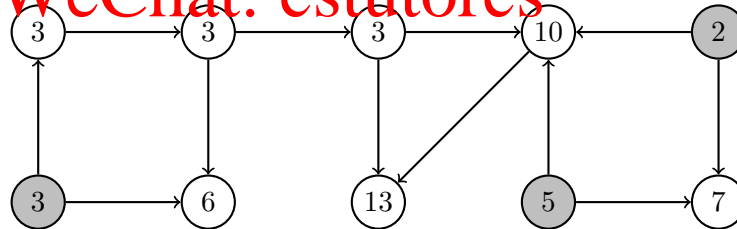
Answers that describe an algorithm should always include a proof of correctness and running time analysis. Algorithms may be described in English or in clear pseudocode with explanatory comments.

1. You are given an integer  $r \in [1..n]$  and a sequence  $\sigma = s_1, s_2, \dots, s_n$  of  $n$  distinct elements in which elements are presented one at a time. When element  $s_i$  is presented, you can no longer access any of  $s_1, \dots, s_{i-1}$  unless your algorithm has stored them. You are asked to output the  $r^{\text{th}}$  smallest element in  $\sigma$ . Design an algorithm that can accomplish this using  $O(r)$  space and  $O(n)$  expected time. **(25 points)**
2. You are given the adjacency-list representation of a directed acyclic graph  $G = (V, E)$  with  $n$  vertices and  $m$  edges. Furthermore, for every vertex  $v \in V$  with in-degree zero, you are given a number  $value(v)$ . For every vertex  $v \in V$  with positive in-degree, we define

$$value(v) = \sum_{(u,v) \in E} value(u),$$

the sum of the values of all of  $v$ 's predecessors in the graph.

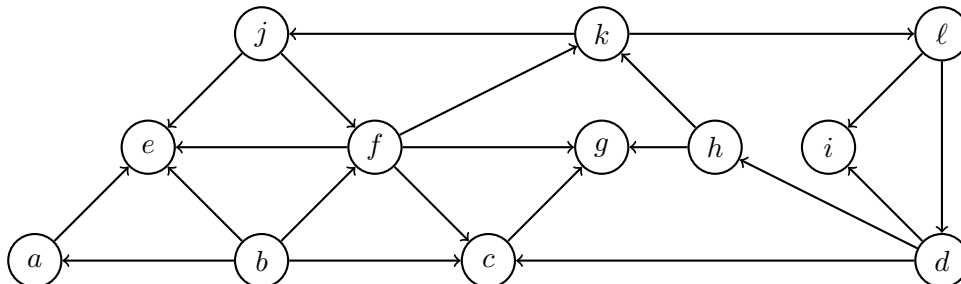
For example, in the graph below, the gray vertices have in-degree zero, so their values determine the values of all other vertices in the graph.



Design a  $O(n + m)$ -time algorithm to compute  $value(v)$  for all vertices  $v \in V$ . **(25 points)**

3. You are given the adjacency list representation of a directed graph  $G = (V, E)$  with  $n$  vertices and  $m$  edges. We define the set  $S = \{v \in V : \text{some cycle in } G \text{ is reachable from } v\}$ .

For example, in the graph below,  $S = \{b, d, f, h, j, k, \ell\}$ .



Design an  $O(n + m)$  time algorithm to find the set  $S$ .

**(25 points)**

4. Let  $G(V, E)$  be an undirected graph with  $n$  vertices and  $m$  edges such that  $G$  has two vertices  $s$  and  $t$  where the shortest path from  $s$  to  $t$  has length strictly more than  $n/2$ . Then, prove that there must be a vertex  $w \in V \setminus \{s, t\}$  such that every path from  $s$  to  $t$  must pass through  $w$ . Furthermore, assuming that  $G$  is given to you in the adjacency-list representation along with vertices  $s$  and  $t$ , design an  $O(n + m)$  time algorithm to output such a vertex  $w$ . **(25 points)**

**Assignment Project Exam Help**

**<https://tutorcs.com>**

**WeChat: cstutorcs**