CSE 431/531: Algorithm Analysis and Design Spring 2022

Homework 1

Instructor: Shi Li **Deadline: 2/23/2022**

Your Name: _____ Your Student ID: _____

Problems	1	2	3	Total
Max. Score	20	25	35	80
Your Score				

Problem 1. For each pair of functions f and g in the following table, indicate whether $f = O(g), f = \Omega(g)$ and $f = \Theta(g)$ respectively.

$f(n)$	$g(n)$	O	Ω	Θ
$\log_2 n$	$5\log_2(n^3) + 3$			
$10n^2 - n$	$n^2 \log n$			
$n^3 - 4n^2 + 10$	n^2			

Prove $\lceil 10n\sqrt{n} \rceil + \lceil n \log n \rceil = O(n\sqrt{n})$.

Problem 2. Consider the following algorithm for sorting an array A of n numbers.

Algorithm 1 Sorting the integer array A, which is of size n

```
1: for i ← 1 to n − 1 do
2:     for j ← i + 1 to n do
3:         if A[i] > A[j] then t ← A[i], A[i] ← A[j], A[j] ← t
```

(2a) What does the pseudo-code "$t \leftarrow A[i], A[i] \leftarrow A[j], A[j] \leftarrow t$" do?

(2b) What is the running time of the algorithm? Briefly explain why. Your bound should be tight (that is, "the running time is $O(n^{10})$" is not considered as a correct answer).

(2c) Why is the algorithm correct? To answer the question, you just need to describe the property that the array A satisfies after each iteration i of the outer loop.

Problem 3. We are given a directed graph $G = (V, E)$ with $|V| = n$ and $|E| = m$, using the linked-list representation. You need to design an $O(n + m)$-time algorithm to decide between the following three cases:

(i) there is no topological-ordering for G, in which case your algorithm should output "none",

(ii) there is a unique topological-ordering for G, in which case your algorithm should output "unique", and

(iii) there are at least two different topological orderings for G, in which case your algorithm should output "multiple".

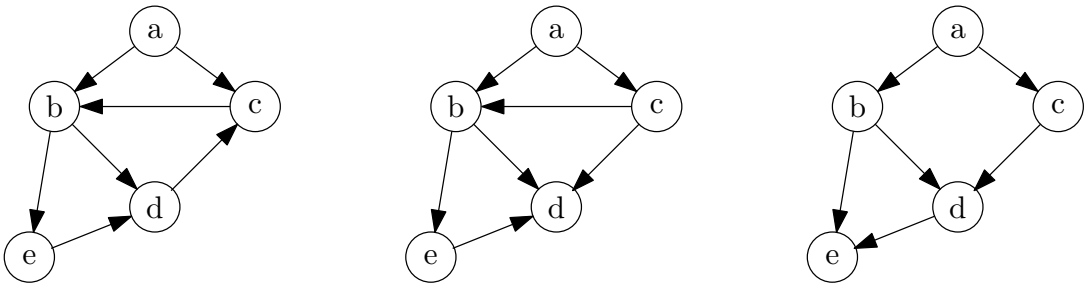

Figure 1: Example input graphs for Problem 3.

For example, consider the three graphs in Figure 1. The outputs for the left-side, middle and right- side graphs are respectively "none", "unique" and "multiple": There is no topological ordering for the left-side graph, there is a unique topological ordering (a, c, b, e, d) for the middle graph, and there are two different topological orderings (a, b, c, d, e) and (a, c, b, d, e) for the right-side graph.

Giving a pseudo-code for your algorithm is sufficient, if the correctness and running time can be easily seen.