

Algorithm HW8

1. Exercise 22.1-1

Given an adjacency-list representation of a directed graph, how long does it take to compute the out-degree of every vertex? How long does it take to compute the in-degrees?

2. Exercise 22.1-3

The **transpose** of a directed graph $G = (V, E)$ is the graph $G^T = (V, E^T)$, where $E^T = \{(v, u) \in V \times V : (u, v) \in E\}$. Thus, G^T is G with all its edges reversed. Describe efficient algorithms for computing G^T from G , for both the adjacency-list and adjacency-matrix representations of G . Analyze the running times of your algorithms.

3. Exercise 22.1-5

The **square** of a directed graph $G = (V, E)$ is the graph $G^2 = (V, E^2)$ such that $(u, v) \in E^2$ if and only if G contains a path with at most two edges between u and v . Describe efficient algorithms for computing G^2 from G for both the adjacency-list and adjacency-matrix representations of G . Analyze the running times of your algorithms.

4. Exercise 22.2-7

There are two types of professional wrestlers: “babyfaces” (“good guys”) and “heels” (“bad guys”). Between any pair of professional wrestlers, there may or may not be a rivalry. Suppose we have n professional wrestlers and we have a list of r pairs of wrestlers for which there are rivalries. Give an $O(n + r)$ -time algorithm that determines whether it is possible to designate some of the wrestlers as babyfaces and the remainder as heels such that each rivalry is between a babyface and a heel. If it is possible to perform such a designation, your algorithm should produce it.

5. Exercise 22.4-2

Give a linear-time algorithm that takes as input a directed acyclic graph $G = (V, E)$ and two vertices s and t , and returns the number of simple paths from s to t in G . For example, the directed acyclic graph of Figure 22.8 contains exactly four simple paths from vertex p to vertex v : pov , $poryv$, $posryv$, and $psryv$. (Your algorithm needs only to count the simple paths, not list them.)

6. Exercise 22.4-3

Give an algorithm that determines whether or not a given undirected graph $G = (V, E)$ contains a cycle. Your algorithm should run in $O(V)$ time, independent of $|E|$.

7. Give an algorithm that determines whether or not a given undirected graph $G = (V, E)$ contains **an euler circuit**, and if the answer is positive your algorithm should output an euler circuit.