The University of Newcastle
School of Electrical Engineering and Computer Science

COMP2230/6230 Algorithms

Tutorial Week 6

23rd – 27th August 2021

Tutorial

1. Using the graph in Figure 1, list the order in which depth-first search visits the vertices, assuming that the vertices are listed in ascending order, starting the search at 1.

2. Using the graph in Figure 1, list the order in which breadth-first search visits the vertices, assuming that the vertices are listed in ascending order, starting the search at 1.

3. Using the directed acyclic graph in Figure 2, trace Algorithm 4.4.1 (Topological Sort).

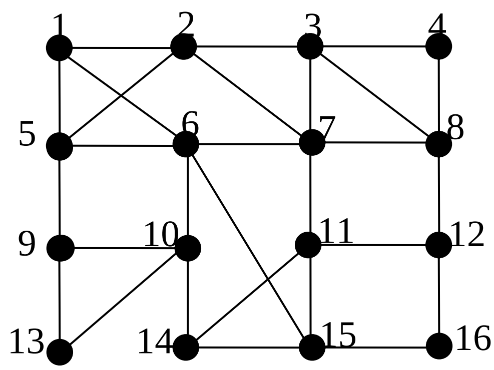

Figure 1: A Simple Graph

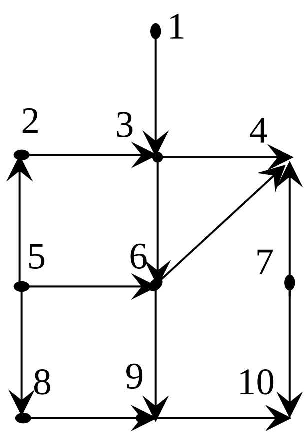

Figure 2: A Directed Acyclic Graph

4. A shortest path $\delta(u, v)$ between two vertices, u and v, in a simple graph has a property that for any vertex k not on the path $\delta(u, v) \leq \delta(u, k) + \delta(k, v)$. That is, the shortest path is the minimum number of edges that have to be traversed to get from u to v (and vice versa in a simple undirected graph). Use either depth-first or breadth-first search to calculate the shortest paths from a fixed start vertex s.

5. Write a backtracking algorithm that prints all permutations of the numbers $1, . . . , n$.

6. Let T [0, 1, . . . , n-1] be a sorted array of distinct integers (they may be negative). Give an algorithm that finds an index i such that T [i] = i, if it exists. What is your algorithm's running time?

Homework

7. Write a non-recursive version of depth-first search.

8. What is the best case time for depth-first search?

9. Show that Algorithm 4.4.1 (Topological Sort) runs in time $\Theta(|V| + |E|)$ for a graph G = (V , E).

10. Give a directed acyclic graph with at least 4 vertices that has a unique topological sort of the graph.

11. Write a backtracking algorithms that prints all subsets of the set {1, 2, . . . , n}.

Extra Questions

12. Write a version of depth-first search in which the input graph is in the form of an adjacency matrix. What is the worst case time of your algorithm?

13. Give an example of a directed acyclic graph with at least 4 vertices in which every permutation of the vertices is a topological sort of the graph.

14. Show all solutions to the 4-Queens problem.

15. Trace a depth-first and breadth-first search for the graphs in exercises 4.2.1 and 4.2.2 in the text.

16. Trace a depth-first and breadth-first search for the directed graphs in exercises 4.4.1, 4.4.2 and 4.4.3 in the text.