ADS 2021: Week 7 Exercises

Exercises for week 7 of Algorithms and Data Structures at ITU. The exercises are from *Algorithms, 4th Edition* by Robert Sedgewick and Kevin Wayne unless otherwise specified. Color-coding of difficulty level and alterations to the exercises (if any) are made by the teachers of the ADS course at ITU.

4.1.1 - Green What is the maximum number of edges in a graph with V vertices and no parallel edges? What is the minimum number of edges in a graph with V vertices, none of which are isolated?

4.2.1 - Green What is the maximum number of edges in a digraph with V vertices and no parallel edges? What is the minimum number of edges in a digraph with V vertices, none of which are isolated?

4.1.28 - Green Two graphs are isomorphic if there is a way to rename the vertices of one to make it identical to the other. Draw all the nonisomorphic graphs with two, three and four vertices.

4.2.8 - Green Draw all the nonisomorphic DAGs with two, three and four vertices.

4.1.12 - Yellow What does the BFS tree tell us about the distance from v to w in an (undirected) graph when neither is at the root?

4.1.16 - Yellow Describe how to find:

a) The eccentricity of a vertex (the length of the shortest path from that vertex to the furthest vertex from it)

b) The diameter of a graph (the maximum eccentricity of any vertex in the graph)

c) The radius of a graph (the smallest eccentricity of any vertex in the graph)

d) A center of a graph (a vertex whose eccentricity is the radius of the graph)

4.1.21 - Yellow Show a trace of the `Bipartite` traversal (p. 547) of the graph described by `tinyGex2.txt` (p. 558). (If you insist, you can do it as detailed as in the figures in the book, but that takes a lot of time.)

Then, describe an alternative implementation that uses BFS instead of DFS. Show a trace of that traversal as well. Are the resulting colourings the same? Are the running times the same? Are there other differences?

4.1.32 - Red Describe a linear-time algorithm to count the parallel edges in a graph.

4.1.36 - Red A bridge in a graph is an edge that, if removed, would separate a connected graph into two disjoint subgraphs. A graph that has no bridges is said to be edge connected. Design and describe two algorithms for determining whether a given graph is edge connected, one using a DFS-based data type and one using a BFS-based data type. Both should run in $O(E \cdot (V + E))$.