The following exercises, taken from CLRS if not otherwise specified, are very good practise.

Practice Exercises

1. Solve Exercises 25.2-1, -2, -6, -8
2. Solve Exercises 26.1-1, -2, -3, -4, -5, -6, -7, -9
3. Solve Exercises 26.2-1, -2, -3, -4, -5, -6, -7, -8, -9, -10
4. Solve Exercises 26.3-1, -2, -3, -4, -5
5. Solve Exercises 33.1-1, -2, -3, -4, -5, -6, -7, -8
6. Solve Exercises 33.2-1, -2, -3, -4, -5, -6, -7, -8, -9
7. Solve Exercises 33.3-1, -2, -3, -4, -5, -6

Problems to hand in

1. (BFS) Describe an algorithm that uses BFS to determine whether or not a given undirected graph is bipartite\(^1\).
   (i) Prove correctness of your algorithm.
   (ii) Analyze running time of your algorithm.

2. (DFS) Modify the DFS algorithm so that it determines the presence of cycles in an undirected graph \(G = (V, E)\) and runs in time \(O(|V|)\), independent of \(|E|\).
   (i) Prove correctness of your algorithm.
   (ii) Explain why it runs in \(O(|V|)\).

3. The edge connectivity of a graph is the minimum number of edges that has to be removed to disconnect the graph. For example, the edge connectivity of a tree is 1, and the edge connectivity of a cyclic chain is 2. Show how the edge connectivity of an undirected graph \(G = (V, E)\)

\(^1\)i.e. it is possible to colour it’s vertices in such a manner that two adjacent nodes have different color, and only two colours are used for all nodes.
can be determined by running a max-flow algorithm on at most $|V|$ flow networks, each having $O(|V|)$ vertices and $O(|E|)$ edges.

4. (i) Write pseudocode to sort a sequence $\langle p_1, \ldots, p_n \rangle$ of $n$ points according to their polar angles with respect to a given origin point $p_0$. Your algorithm should run in time $O(n \log n)$ and use cross-product to compare angles.

(ii) Show how to determine in $O(n^2 \log n)$ time whether any three points in a set of $n$ points are collinear.

(iii) Give an $O(n \log n)$-time algorithm to determine whether an $n$-vertex polygon is simple\(^2\), but not necessarily convex.

\[^2\text{i.e. do not cross itself}\]