This is a brief recap of prerequisites from discrete maths and data structures. If you need to refresh your memory, Chapter 0 of Sipser’s book has a nice summary.

1. Prove the set theoretic identity \((A \cap B) \times (C \cap D) = (A \times C) \cap (B \times D)\).

2. Let \(a \neq 0\) be some fixed real number
   
   (a) Give a recursive definition for the power \(a^n\), where \(n \in \mathbb{N}\).
   
   (b) Prove by induction that the identity \(a^n \cdot a^k = a^{n+k}\) holds for all \(n, k \in \mathbb{N}\).

3. A relation \(\sim\) on a set \(X\) is called an equivalence relation if it has the following three properties:
   
   - reflexive: \(x \sim x\) for all \(x \in X\)
   - symmetric: for all \(x, y \in X\), if \(x \sim y\), then \(y \sim x\)
   - transitive: for all \(x, y, z \in X\), if \(x \sim y\) and \(y \sim z\), then \(x \sim z\).
   
   (a) Let \(G = (V, E)\) be an undirected graph. Write \(u \sim v\), if there is a path from vertex \(u\) to vertex \(v\). Is \(\sim\) an equivalence relation on \(V\)? Explain briefly.
   
   (b) Let \(G = (V, E)\) be a directed graph. Write \(u \sim v\), if there is a path from vertex \(u\) to vertex \(v\). Is \(\sim\) an equivalence relation on \(V\)? Explain briefly.

4. Give the (a) depth-first tree and (b) breath-first tree for the graph below. You don’t need to show any intermediate stages, just the trees. In situations where the algorithm can choose between several vertices, assume they are picked in alphabetical order.