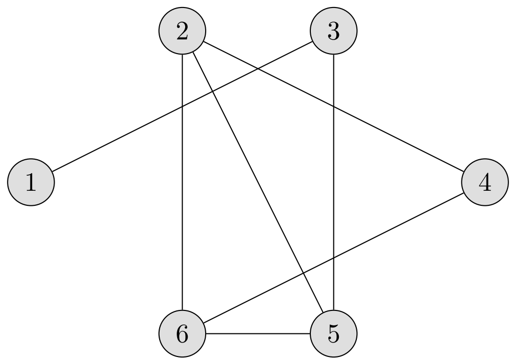
Exercise 1 Simple training exercises

1. Give an adjacency-list representation for a complete binary tree on 7 vertices. Give an equivalent adjacency-matrix representation. Assume that the edges are undirected and that the vertices are numbered from 1 to 7 as in a binary heap. (CLRS 20.1-2)
2. Show the d and π values that result from running breadth-first search on the directed graph of Figure 20.2(a), using vertex 3 as the source. (CLRS 20.2-1)
3. Show the d and π values that result from running breadth-first search on the undirected graph of Figure 20.3, using vertex u as the source. Assume that neighbours of a vertex are visited in alphabetical order. (CLRS 20.2-2)

Exercise 2 Question from Re-exam 2024

Consider the following undirected graph G:



1. Show the adjacency list representation of the graph (sort the adjacency lists by the value in the node).
2. Use BFS (CLRS 4th edition Chapter 20.2, CLRS 3rd edition Chapter 22.2) on G with node 1 as source (ie. BFS(G, 1)) and show the graph after exactly 4 iterations of the **while**-loop on line 10. That is, annotate each node with the distance found and color each node appropriately (either white, gray or black).

Exercise 3 Fun creative exercises!

1. What is the running time of BFS if we represent its input graph by an adjacency matrix and modify the algorithm to handle this form of input? (CLRS 20.2-4)
2. Argue that in a breadth-first search, the value $u.d$ assigned to a vertex u is independent of the order in which the vertices appear in each adjacency list. Using Figure 20.3 as an example, show that the breadth-first tree computed by BFS can depend on the ordering within adjacency lists. (CLRS 20.2-5)
3. Rewrite the procedure DFS, using a stack to eliminate recursion. (CLRS 20.3-6)