University of Houston

Homework 3 Solutions

COSC 3320
Algorithms and Data Structures

Due: Thursday, April 11, 2024
11:59 PM

Note

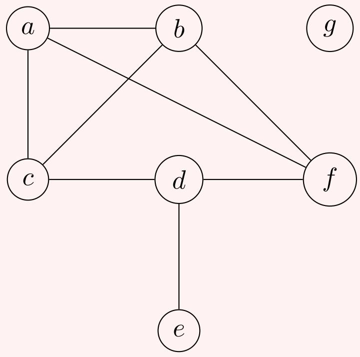
Read the Academic Honesty policy.

The below material is for the use of the students enrolled in this course only. This material should not be further disseminated without instructor permission. This includes sharing content to commercial course material suppliers such as Course Hero or Chegg. Students are also prohibited from sharing materials derived from this content.

1 Exercises

Exercise 1: Graph Traversal (20 Points)

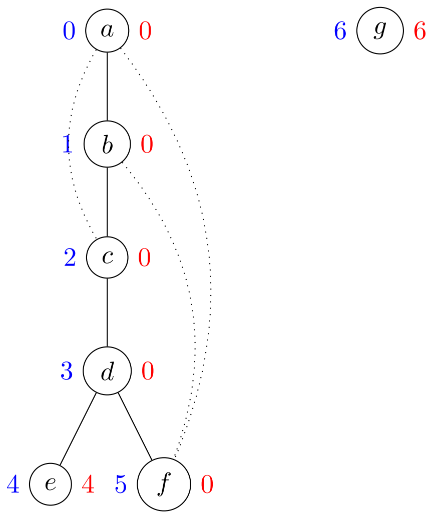
Consider the graph G given below.



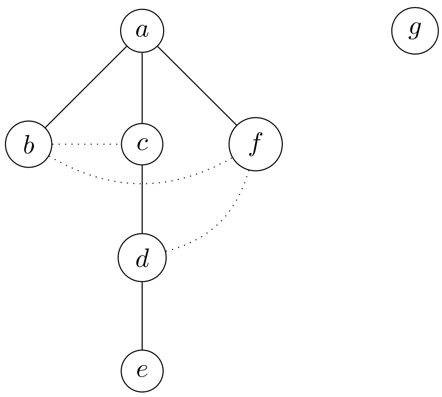
1. Draw the DFS forest of G starting from vertex a. Label the nodes with their DFS numberings and `low` numberings. Draw back edges as dotted lines. Break ties alphabetically.
2. Draw the BFS forest of G starting from vertex a. Break ties alphabetically.
3. List the connected components of G.
4. List the cut vertices of G.
5. List the biconnected components of G.
6. List the cut edges of G.

Solution.

1.



2.



3. $\{a, b, c, d, e, f\}$, $\{g\}$.
4. The only cut vertex is d.
5. The biconnected components are $\{a, b, c, d, f\}$, $\{d, e\}$, and $\{g\}$.
6. The only cut edge is (d, e).

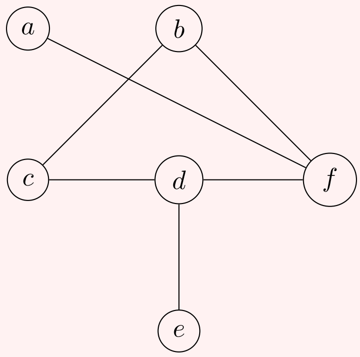
$\square$

Exercise 2: Bipartite (20 Points)

An undirected graph $G = (V, E)$ is called *bipartite* if it contains no *odd* cycle, i.e., no cycle has an odd number of edges. Note that a graph with no cycles is also bipartite. We are going to show an important property of bipartite graphs: its vertex set can be partitioned into two disjoint sets A and B, i.e., $A \cup B = V$ and $A \cap B = \phi$, such that all the edges are between vertices in A and vertices in B — in other words, there are no edges between any two vertices in A and any two vertices in B.

The goal of this problem is that, given a *connected bipartite* graph $G = (V, E)$, to find a partition of the vertex set into A and B such that all edges are between some vertex in A and some vertex in B.

Consider the bipartite graph below:



1. List all cycles of the graph.
2. Give sets A and B for the graph.
3. Show that if G is bipartite then there are no edges between vertices in the same level of the BFS tree.
4. Show that you can output the sets A and B in linear time, i.e., $\mathcal{O}(|V| + |E|)$ time.

Solution.

1. The only cycle is $\{b, c, d, f\}$.
2. $$A = \{a, b, d\}$$ $$B = \{c, e, f\}$$
3. By way of contradiction, assume there exist two nodes, u and v, in the same level of the BFS tree that have an edge between them. Since they are on the same level, they must have a common ancestor, say r. Additionally, the number of edges in the path from r to u is the same as that from r to v — call this value ℓ. Then, the path formed by joining the path from r to v, then edge (u, v), then the path from v to r, is a cycle of length $2\ell + 1$, which is odd. This contradicts that the graph is bipartite. Thus, there can be no edge between vertices on the same level of the BFS tree.
4. Simply perform BFS — put the root in A, the neighbors in B, the neighbors neighbors in A, and so on, i.e., put level 0 in A, level 1 in B, etc. Since the graph is bipartite, there are no edges between nodes in the same level — thus, all edges go from A to B.[1]

   (If the graph is not bipartite, a simple modification solves the issue — if you must place a node in A, but it is already in B, or vice-versa, simply output that the graph is not bipartite.)

□

[1]To be precise, we need to show that there exist no edge between nodes u and v in the same subtree if their depths have the same parity. The argument is virtually identical, except the lengths are now ℓ_1 and ℓ_2. However, since their parity is the same, $\ell_1 + \ell_2$ is even, and we can construct the odd cycle.

Exercise 3: Connected Components (20 Points)

Solve the problem Number of Islands at LeetCode:

1. using DFS
2. using BFS

Submit links to the submission results for each.