Jonas Hemmett 9/27/25

(1) Paths in a graph

Suppose we have a connected graph G = (V, E) having n > 0 nodes and m > 0 edges. Answer the following questions:

(1a) What is the longest simple path that G can have? Explain your answer. (Recall: a simple path does not have any cycles.) [two points]

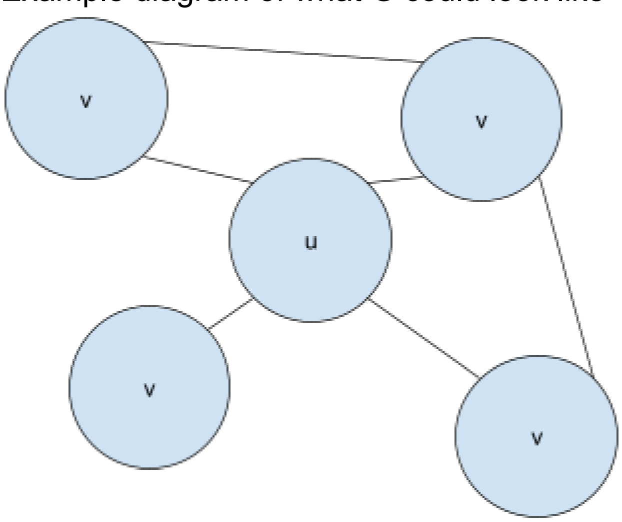
Since there are no cycles, all nodes can be visited at most 1 time. This means we could go to at most n nodes. A path with n nodes has n - 1 edges, so the longest possible path is n - 1.

(1b) Suppose that the shortest-path distance between any two nodes u, v ∈ V is the same. Without making any assumptions about how many nodes or edges G has, what structure must G have? Explain your answer. [two points]

Since G is connected, there must exist a pair of nodes such that (u, v) ∈ E. Since an edge exists, there exists a path along that edge with a distance of one. Since all nodes have the same shortest distance, all node pairs must have a distance of one. In other words, every node pair has an edge (u, v) ∈ E by definition making G a complete graph.

(1c) Suppose that there is a node u ∈ V such that the shortest-path distance from u to any other node of G is one. What does this then imply about the shortest-path distance between any other two nodes of G? Explain your answer. [two points]

Example diagram of what G could look like

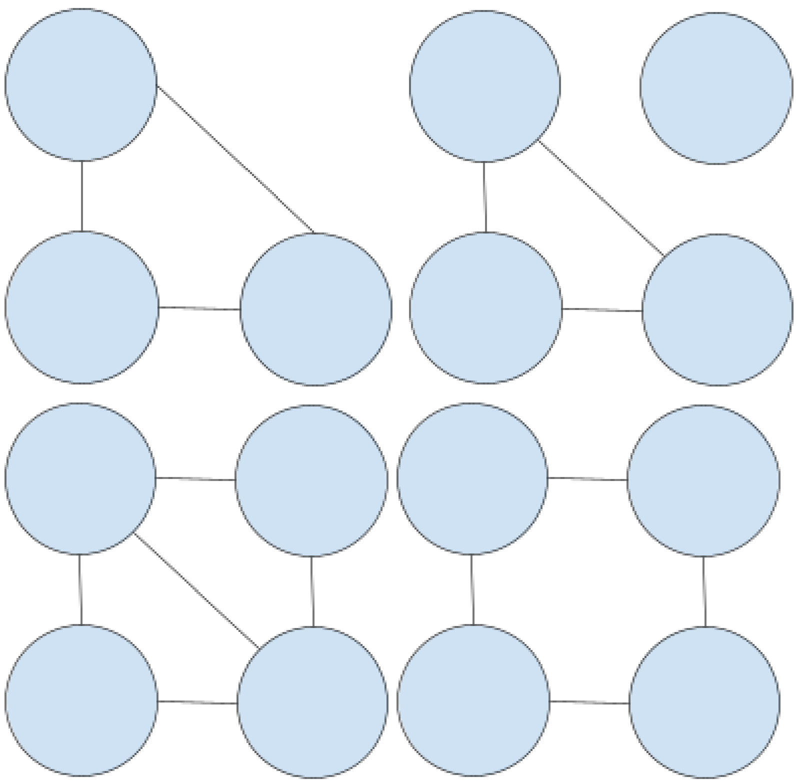


Let V' = {n | n ∈ V ∧ n ≠ u}. for any two nodes v, w ∈ V', we know v and w both have an edge with u so (v, u) (u, w) ∈ E. These two edges form the path (v, u, w) with a distance of two. However it is also possible that (v, w) ∈ E creating a path with a distance of one. So the shortest-path distance between any other two nodes of G is in the range [1, 2].

(2) Edges in a graph

Suppose that in a connected graph G = (V, E), every node has an edge to two other nodes, and these are the only edges that the node has. What is the number of edges in the graph, m, as a function of the number of nodes, n? Prove this inductively. [four points]

Example diagrams



I propose that m = n.

Base case: For every node to connect to two other nodes there must be a minimum of three nodes in G. $u_1, u_2, u_3 \in V$, $(u_1, u_2), (u_2, u_3), (u_3, u_1) \in E$. Here we can see that there are three edges and three nodes .

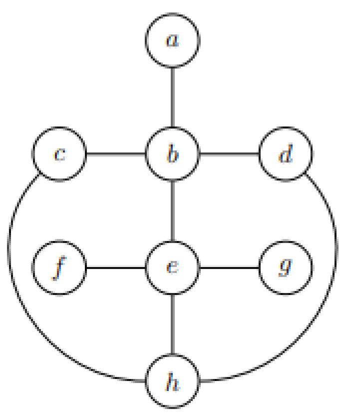
Inductive step: Since we have shown that m = n holds for the base case, we must now prove it holds for a graph with k + 1 nodes. First a graph with k nodes would have $u_1, u_2, \ldots u_k \in V$ and $(u_1, u_2), (u_2, u_3), \ldots (u_k, u_1) \in E$ for some combination of node pairs. We can see that each node forms an edge with the following node and the last node connects back to the first node.

Now for k + 1 we would have $u_1, u_2, \dots u_k, u_{k+1} \in V$. Now update the edges, right now we have $(u_1, u_2), (u_2, u_3), \dots(u_k, u_1)$ but u_{k+1} has zero edges instead of two. We start by adding an edge going from u_{k+1} to any other node v. We then add another edge going from u_{k+1} to another node w such that $(v, w) \in E$. Here u_1 and u_k are being using for v and w respectively, we now have $(u_1, u_2), (u_2, u_3), \dots(u_k, u_1), (u_k, u_{k+1}), (u_{k+1}, u_1)$ but now nodes v and w have a total of three edges instead of two, this can be solved by removing (u, v) from E. We are now left with $(u_1, u_2), (u_2, u_3), \dots(u_k, u_{k+1}), (u_{k+1}, u_1) \in E$ for some combination of node pairs which shows m = k + 1

Since our hypothesis of m = n holds for both the base case and inductive step, we have shown that it is true.

(3) Graph traversal

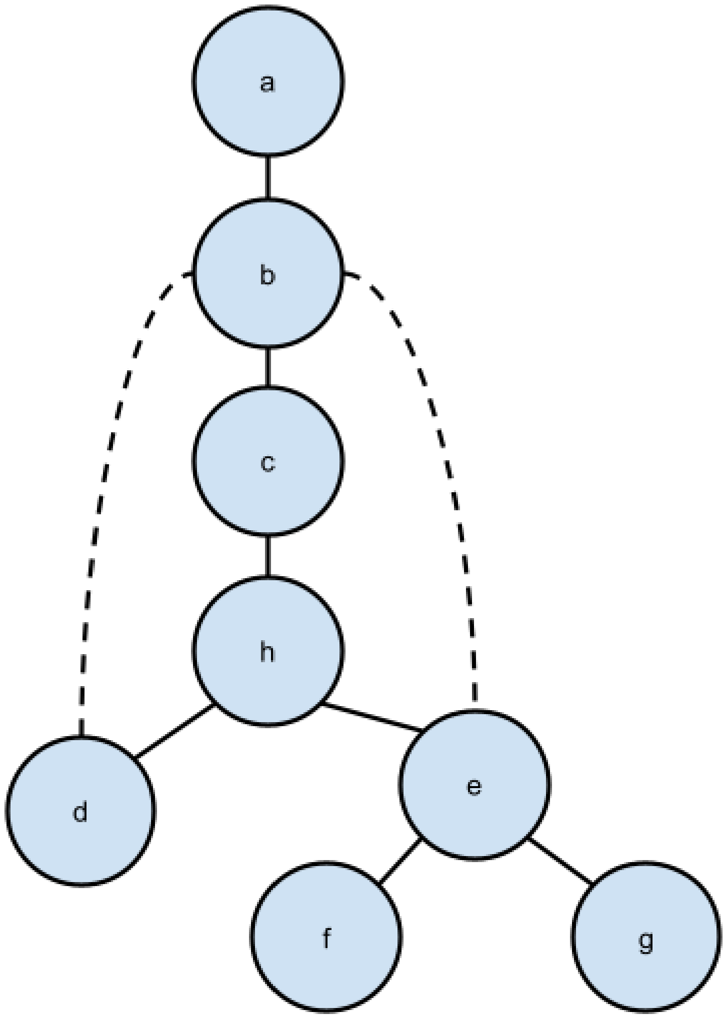
Here is an undirected graph:



3.1 DFS

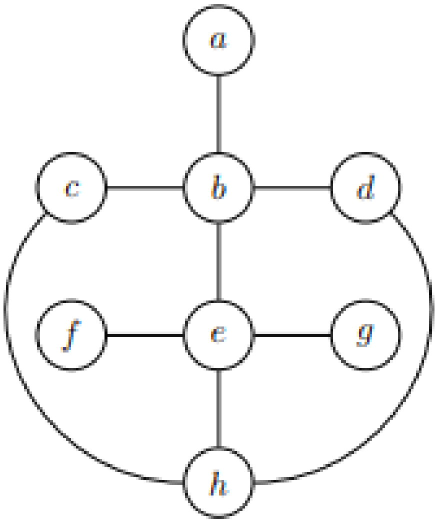
Show the depth-first search tree for this graph if we start the traversal at node a. When there is a choice of nodes to explore, explore the one having the smallest node number lexicographically (i.e., in sorted alphabetic order). Also draw, as dotted lines, the graph edges that are not tree edges. [three points]

At	Stack	Backtrack
	a	
a	b	
b	c, d, e	
c	h, d, e	
h	d, e	
d	e	h
e	f, g	
f	g	e
g		

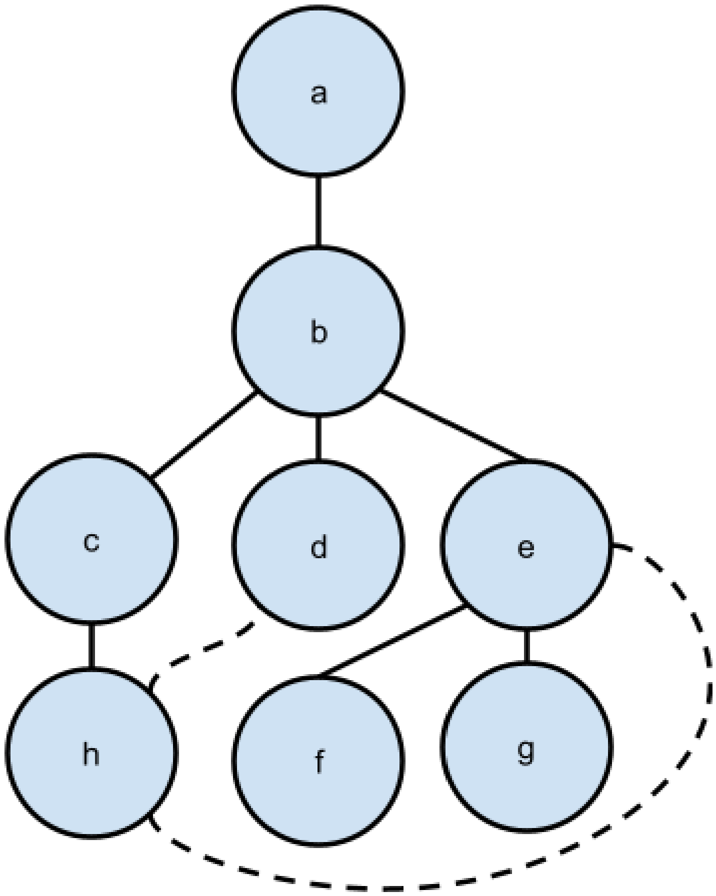


3.2 BFS

Show the breadth-first search tree for this graph if we start the traversal at node a. When there is a choice of nodes to explore, explore the one having the smallest node lexicographically (i.e., in sorted alphabetic order). Also draw, as dotted lines, the graph edges that are not tree edges. [three points]

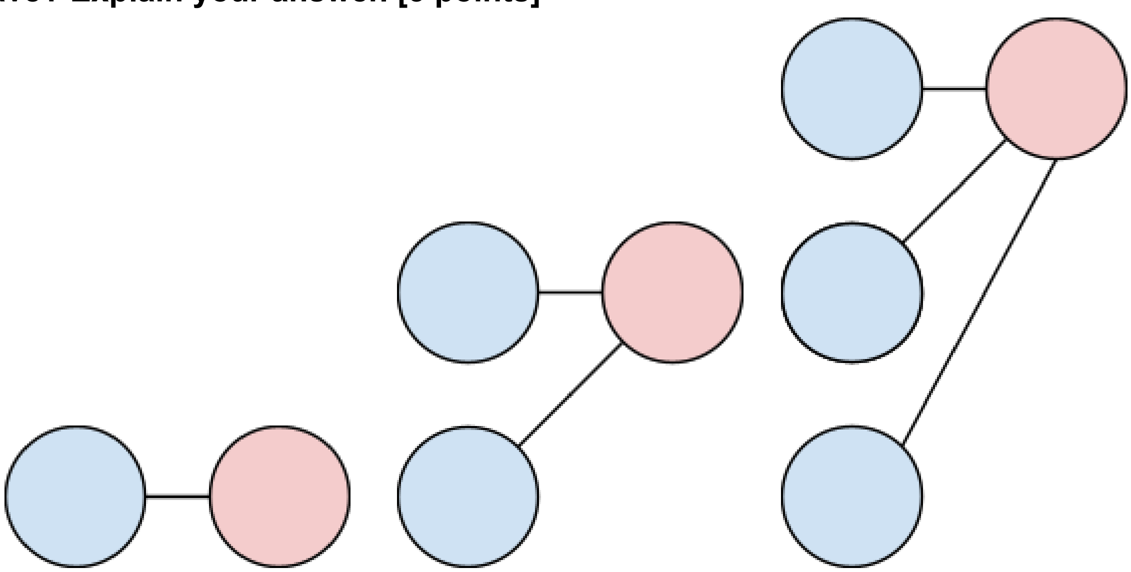


At	Stack
	a
a	b
b	c, d, e
c	d, e, h
d	e, h
e	h, f ,g
h	f, g
f	g
g	



(4) Bipartite graphs

What is the smallest number of edges, m, a connected bipartite graph having n nodes can have? Explain your answer. [3 points]



The smallest number of edges is n - 1.

Proof by induction

Let G be split up into two sets A and B. Let A contain any one node from G node. Let B contain every other node from G. (So $|A| = 1$, $B = G - A$).

Base case: $n = 2$, $u \in A$ and $v \in B$, $(u, v) \in V$. Here we can see that $m = 1$.

Inductive step: We must now prove that if the graph holds true for $n = k$ nodes, it must also hold true for $n = k + 1$ nodes. Let $u \in A$, $v_1, v_2, \ldots, v_k \in B$ and $(u, v_1), (u, v_2), \ldots, (u, v_k) \in V$. Here we can clearly see that when $n = k$, $m = k - 1$.

When we add another node, w, we add it to B and add the edge (w, u) to V. We can see that $n = k + 1$ and $m = k$

We can see that this graph is still bipartite as $w \in B$ and the only edge to w has connected to a node in A

We can see that the graph is still connected because w is connected to u, and all previous nodes are connected to u.