Written Assignment #3

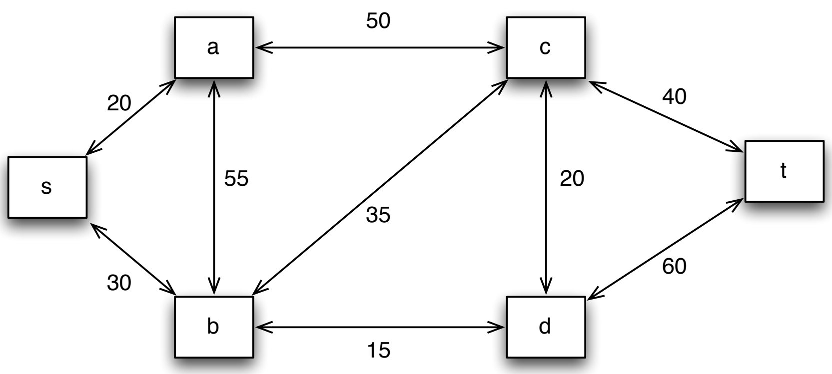
Vancouver Summer Program 2019 – Algorithms – UBC

- You should work with a partner.
- You must typeset your solutions.
- Submit your work using Gradescope by **10:00 p.m. on Tuesday, July 30**.
- **Notation.** $\mathbb{N} = \{1, 2, \dots\} \subset \{0, 1, 2, \dots\} = \mathbb{Z}_+$, and $\mathbb{R}_+ = [0, \infty)$.

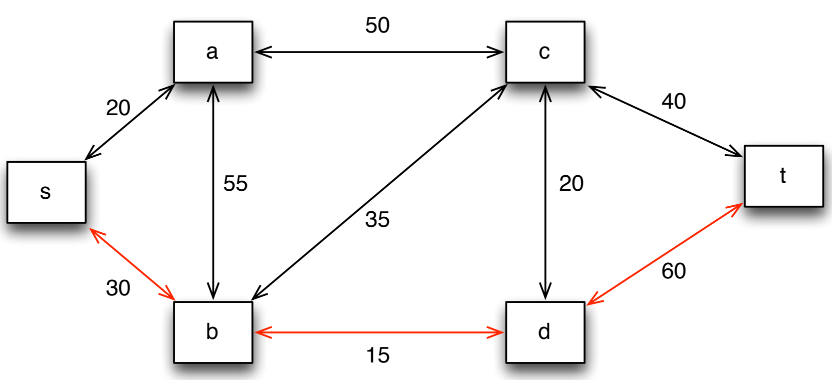
---

1. (Applying graph algorithms: 5 points.)

   (a) (1 point) Find the shortest path between s and t in the following graph.



Solution.



   (b) We have three containers whose sizes are 10 Liters (L), 7 L, and 4 L, respectively. The 7 L and 4 L containers start out full of water, but the 10 L container is initially empty. We are allowed one type of operation: pouring the contents of one container into another, stopping only when the source container is empty or the destination container is full. We want to know if there is a sequence of pourings that leaves exactly 2 L in the 7 or 4 L container.

(i) (2 point) Model this as a graph problem: give a precise definition of the graph involved by clearly explaining the vertices and edges, and state the specific question about this graph that needs to be answered.

Solution. Let $G = (V, E)$ be our (directed) graph. We will model the set of nodes as triples of numbers (a_0, a_1, a_2) where the following relationships hold: Let $S_0 = 10, S_1 = 7, S_3 = 4$ be the sizes of the corresponding containers. a_i will correspond to the actual contents of the ith container. The following must hold: $0 \le a_i \le S$ for $i \in \{0, 1, 2\}$, and at any given node $a_0 + a_1 + a_2 = 11$ (the total amount of water we started from). An edge between two nodes (a_0, a_1, a_2) and (b_0, b_1, b_2) exists if both the following are satisfied:

- the two nodes differ in exactly two coordinates (and the third one is the same in both), and
- if i, j are the coordinates they differ in, then either $a_i = 0$ or $a_j = 0$ or $a_i = S_i$ or $a_j = S_j$.

The question that needs to be answered is whether there exists a path between the nodes $(0, 7, 4)$ and $(*, 2, *)$ or $(*, *, 2)$ where $*$ stands for any (allowed) value of the corresponding coordinate.

(ii) (1 point) What algorithm should be applied to solve the problem?

Solution. Given the above description, it is easy to see that DFS on that graph should be applied, starting from node $(0, 7, 4)$ with an extra line of code that halts and answers `yes` if one of the desired nodes is reached and `no` if all the connected component of the starting node is exhausted an no desired vertex is reached.

(iii) (1 point) Find the answer by applying the algorithm.

Solution. After a few steps of the algorithm (depth 6 on the dfs tree) the node $(2, 7, 2)$ is reached, so we answer `yes`.

2. (Some simple graph properties: 5 points.) Let G be a graph with v vertices and e edges. Let M be the maximum degree of the vertices of G, and let m be the minimum degree of the vertices of G. Which of the following propositions must be true? Provide a short proof or counterexample in each case.

(a) (1 point) $2e/v \le M$.

Solution. True. If $v \ge 1$, then $2e/v = \left(\sum_{x \in V} \deg(x)\right)/v \le vM/v = M$.

(b) (1 point) $2e/v \ge m$.

Solution. True. If $v \ge 1$, then $2e/v = \left(\sum_{x \in V} \deg(x)\right)/v \ge vm/v = m$.

(c) (1 point) There exists a simple path (includes no cycles) of length at least m.

Solution. True. Suppose that the longest path in G has length $k < m$. Consider one such path, say $P = x_1 \cdots x_{k+1}$ for some labeling $x_1, \ldots, x_v$ of the vertices. Consider x_{k+1}. Then x_{k+1} cannot be connected to any vertices outside P, because this will result in a longer path, contradicting the fact that P is a longest path. Since $\deg(x_{k+1}) \ge m$, it follows that, by the preceding argument, x_{k+1} must be connected to at least m vertices in $P - x_{k+1}$; this is impossible, because $P - x_{k+1} \equiv x_1 \cdots x_k$ contains $k < m$ vertices.

(d) (1 point) $m > 2$ implies that G is connected.

Solution. False. Consider the graph consisting of two components, each of which is a "polygon" (i.e., a cycle) of 5 sides (edges), with an additional center node that is connected by an edge to every node on the polygon. This graph has $m = 3$ but is not connected.
Also any graph consisting of two or more components, where each component is a complete graph on $v \ge 4$ vertices is a counterexample.

(e) (1 point) In every (simple) graph there is a path from any vertex of odd degree to some other vertex of odd degree.

Solution. There are many ways to show this; here's one. Let u be a vertex of odd degree in G. There must be a component (i.e., a maximal connected subgraph) containing u, so take that component. Thus we may assume without loss of generality that G is connected. Recall that the number of vertices of odd degree in any (simple) graph must be even. Since u is of odd degree, there must exist a vertex $v \in G$ whose degree is also odd. There is a path between u and v because they are in the same component, and we are done.

3. (BONUS. Some more graph properties: 2 points.) Let m be a positive integer and consider a graph G^* with $2m$ vertices: $v_1, \ldots, v_{2m}$. An edge exists between vertices v_i and v_j if and only if $(i - j \equiv 1 \mod 2m) \vee (i - j \equiv 2m - 1 \mod 2m) \vee (i - j \equiv m \mod 2m)$.

   Note that $x \equiv y \mod n$ if and only if $x = kn + y$ for some integer k. As examples, $25 \equiv 5 \mod 20$, $29 \equiv -1 \mod 30$ and $29 \equiv 29 \mod 30$.

   (a) (1 point) For each $j \in \{2, \ldots, 2m\}$, what is the distance between v_1 and v_j? The *distance* between two vertices of a graph is the number of edges on the shortest path that connects the two vertices. (Derive an expression in terms of i, j and m. You will have to consider a few cases.)

   Solution. If $j \leq \lfloor m/2 \rfloor + 1$, the distance is $j - 1$. If $\lfloor m/2 \rfloor + 1 < j \leq m + 1$, the distance is $m - j + 2$. If $m + 1 < j \leq \lceil 3m/2 \rceil$, the distance is $j - m$. If $\lceil 3m/2 \rceil < j \leq 2m$, the distance is $2m - j + 1$.

   (b) (1 point) A graph G is k-edge-connected if and only if one has to remove k edges to disconnect the graph. Prove that G^* is not 4-edge-connected: you can remove three or fewer edges to disconnect the graph.

   Solution. Consider v_1. If $m = 1$, remove the single edge to disconnect the graph; otherwise remove the three edges incident on v_1.