

Some practice counting

Question 1. Let $G = (V, E)$ be an undirected graph with n vertices and m edges. Show that if every vertex has degree at least $n/2$, the graph is connected. You may assume that n is even. As a hint, think about cuts in the graph. A cut (S, T) is a partition of the vertices into two sets S and T such that $S \cup T = V$ and $S \cap T = \emptyset$.

Question 2 (KT 3.9). There's a natural intuition that two nodes that are far apart in a communication network—separated by many hops—have a more tenuous connection than two nodes that are close together. There are a number of algorithmic results that are based to some extent on different ways of making this notion precise. Here's one that involves the susceptibility of paths to the deletion of nodes.

Suppose that an n -node undirected graph $G = (V, E)$ contains two nodes s and t such that the distance between s and t is strictly greater than $n/2$. Show that there must exist some node v , not equal to either s or t , such that deleting v from G destroys all $s - t$ paths. (In other words, the graph obtained from G by deleting v contains no path from s to t .) Give an algorithm with running time $O(m + n)$ to find such a node v . Describe your algorithm in prose and prove that it works correctly. Make sure your algorithmic description is clear and concise. As a hint, imagine performing a breadth-first search from s . How large is each layer L_i along the way to t ?

Algorithms in the wild

Question 3 (KT 3.8). A number of stories in the press about the structure of the Internet and the Web have focused on some version of the following question: How far apart are typical nodes in these networks? If you read these stories carefully, you find that many of them are confused about the difference between the diameter of a network and the average distance in a network; they often jump back and forth between these concepts as though they're the same thing.

As in the text, we say that the distance between two nodes u and v in a graph $G = (V, E)$ is the minimum number of edges in a path joining them; we'll denote this by $\text{dist}(u, v)$. We say that the diameter of G is the maximum distance between any pair of nodes; and we'll denote this quantity by $\text{diam}(G)$.

Let's define a related quantity, which we'll call the average pairwise distance in G (denoted $\text{apd}(G)$). We define $\text{apd}(G)$ to be the average, over all $\binom{n}{2}$ sets of two distinct nodes u and v , of the distance between u and v . That is,

$$\text{apd}(G) = \left[\sum_{\{u,v\} \subseteq V} \text{dist}(u, v) \right] / \binom{n}{2}$$

Here's a sample example to convince yourself that there are graphs G for which $\text{diam}(g) \neq \text{apd}(G)$. Let G be a graph with three nodes u, v, w , and with two edges $\{u, v\}$ and $\{v, w\}$. Then

$$\text{diam}(G) = \text{dist}(u, w) = 2,$$

while

$$\text{apd}(G) = [\text{dist}(u, v) + \text{dist}(u, w) + \text{dist}(v, w)]/3 = 4/3.$$

Of course, these two numbers aren't all that far apart in the case of this three-node graph, and so it's natural to ask whether there's always a close relation between them. Here's a claim that tries to make this precise.

Claim 1. There exists a positive natural number c so that for all connected graphs G , it is the case that

$$\frac{\text{diam}(G)}{\text{apd}(G)} \leq c$$

This claim is actually false. Show that given a constant $c > 0$ you can always construct a graph G such that $\text{diam}(G)/\text{apd}(G) > c$. As hint, you may find inspiration from star-tipped, magic wands.

Diameters revisited (from a computer scientist's perspective)

Question 4. *In the last question we defined the diameter of a graph, but we did not discuss methods for computing it. Here we will consider trees. Give a linear time, $O(n + m)$ -algorithm to find the diameter of a tree $T = (V, E)$. Prove that your algorithm is correct.*

Question 5. *Does your algorithm in Question 4 extend to undirected graphs? If so, prove it. If not, provide a counterexample.*

Extra credit: Approximating the diameter of a graph

Question 6. *There's a good chance your linear-time algorithm in Question 4 doesn't extend to undirected graphs. If it does, you almost certainly have a STOC publication. Congratulations! It turns out that, until very recently, we didn't have any method for computing the diameter of a graph that didn't first compute the shortest path between all pairs of nodes. When graphs are dense, all-pairs shortest paths is fairly expensive, so some people have explored quicker algorithms which estimate the diameter of the graph. Develop a linear-time algorithm that, given a graph G , returns a diameter estimate that is always within a factor of $1/2$ of the true diameter. That is, if the true diameter is $\text{diam}(G)$ then you should return a value k where $\text{diam}(G)/2 \leq k \leq \text{diam}(G)$.*

Question 7. *I found this assignment:*

1. *Icky. I'd prefer a slow spinal tap.*
2. *Light. C'mon Brent, throw me the deuce!*
3. *Perfect. Like a slow sip of the finest bourbon and bitters.*
4. *Pleasurable, but pushy in a good way.*
5. *Draining. I barely made it. I need sleep. Send in the closer.*
6. *Awful. Donald Knuth couldn't solve these problem, how do you expect me to?*