Discrete Math in CS	**Problem Set 8**
CS 280 Fall 2005	**Due November 18, 2005**

Reading: Rosen, Sections 8.1, 8.2, 8.4, 9.4.

---

(1) *Ad hoc networks*, made up of low-powered wireless devices, have been proposed for situations like natural disasters in which the coordinators of a rescue effort might want to monitor conditions in a hard-to-reach area. The idea is that a large collection of these wireless devices could be dropped into such an area from an airplane, and then be configured into a functioning network.

Suppose we have a collection of 8 such devices (real applications would have a much, much larger number), we drop them onto a region that we'll model as the plane, and they land at the following points as specified by 2-dimensional coordinates:

$$(0,0), (50,0), (50,50), (100,50), (100,100), (100,150), (150,50), (200,50).$$

Furthermore, suppose that two of these devices are able to communicate provided that they are within 80 meters of one another.

(a) Draw a picture of the following graph G: the nodes correspond to the set of 8 devices, and there is an edge joining each pair of devices that can communicate.

(b) Draw a spanning tree of G produced by breadth-first search starting from the node at coordinates $(50,0)$.

(2) A number of recent 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 class, 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 $dist(u,v)$. We say that the *diameter* of G is the maximum distance between any pair of nodes; and we'll denote this quantity by $diam(G)$.

Let's define a related quantity, which we'll call the *average pairwise distance* in G (denoted $apd(G)$). We define $apd(G)$ 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,

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

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

$$diam(G) = dist(u, w) = 2,$$

while

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

Of course, these two numbers aren't all *that* far apart in the case of this 3-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: There exists a positive natural number c so that for all connected graphs G, it is the case that*
>
> $$\frac{diam(G)}{apd(G)} \leq c.$$

Decide whether you think the claim is true or false, and give a proof of either the claim or its negation.

(3) In class, we considered a random process that generates trees, and we considered some of its properties. Here we consider a different random process that generates arbitrary graphs.

The random process is very simple to describe. We start with $n + 2$ nodes, labeled $s, t, v_1, v_2, v_3, \ldots, v_n$. For some parameter $p > 0$, we then construct an edge between each pair of nodes independently with probability p.

There are many properties we could consider about the resulting random graph. Here we'll focus on the question of whether there's a short path between s and t; for example, we can imagine s and t as the source and destination for some data we need to transport, and we want the edges produced to yield an efficient way of getting from s to t.

(a) Let $\mathcal{E}(n, p)$ be the event that there is a path of length exactly 2 connecting s to t, when we generate a graph according to the process above with particular values of n and p. (In other words, we're asking about the existence of a path consisting of the edges (s, v_i) and (v_i, t) for some i.) Give a formula for $\Pr[\mathcal{E}(n, p)]$ in terms of n and p.

(b) By varying the value of p, we can produce graphs with a larger or smaller expected number of edges; and we simultaneously affect $\Pr[\mathcal{E}(n, p)]$, the probability of having a length-2 path between s and t. One natural way to parametrize p is to define $p = n^{-\alpha}$, and then to let α range from 0 to ∞.

We say that a value of α is *critical* if, by setting $p = n^{-\alpha}$, the probability of a length-2 path between s and t converges to a number strictly between 0 and 1 as n goes to infinity. More succinctly, α is critical if

$$\lim_{n \to \infty} \Pr\left[\mathcal{E}(n, n^{-\alpha})\right] = c$$

for some $0 < c < 1$. Such a choice of α is interesting because the probability of a length-2 path doesn't converge to either 0 or 1, but remains somewhere in between even as n grows arbitrarily large.

Give a value of α that is critical, and provide an explanation for your answer.

(4) In class, we argued that if a graph G has the property that every node has degree at least 2, then it must contain a cycle. (Recall that the *degree* of a node is the number of edges that are incident to it. Also, as in class, we allow each pair of nodes v, w to be joined by at most one edge; so if a node has degree d, then it has d distinct neighboring nodes in the graph.)

We now want to claim that if we require the minimum degree to be large, then we can in fact find a long cycle.

First, let's say that a cycle in a graph is *simple* if it doesn't repeat nodes; in other words, it consists of nodes $v_1, v_2, v_3, \ldots, v_k$ in sequence, where $v_1 = v_k$ and $v_1, v_2, \ldots, v_{k-1}$ are all different.

For every $d \geq 2$, prove that if a graph G has the property that every node has degree at least d, then it contains a simple cycle with at least $d + 1$ nodes. (*Hint:* Consider creating a path in G by successively moving along edges from one node to another, trying to avoid repeating nodes for as long as possible. Think about what happens at the end of this process, when you can no longer avoid repeating nodes.)