1. Let (G, s, t) be a flow network with integer capacities. An edge e is *upper-binding* if increasing the capacity on e increases the value of the maximum flow in G.

   Describe and analyze an algorithm that finds all the upper-binding edges in the graph.

2. The NSA has established several monitoring stations around the country, each one conveniently hidden in the back of a Starbucks. Each station can monitor up to 42 cell-phone towers, but can only monitor cell-phone towers within a 20-mile radius. To ensure that every cell-phone call is recorded even if some stations malfunction, the NSA requires each cell-phone tower to be monitored by at least 3 different stations.

   Suppose you know that there are n cell-phone towers and m monitoring stations, and you are given a function $\text{DISTANCE}(i, j)$ that returns the distance between the i-th tower and the j-th station in $O(1)$ time. Describe and analyze an algorithm that either computes a valid assignment of cell-phone towers to monitoring stations, or reports correctly that there is no such assignment (in which case the NSA will build yet another Starbucks).

3. Let G = (V, E) be an arbitrary connected graph with weighted edges.

   (a) Prove that for any partition of the vertices V into two disjoint subsets, the minimum spanning tree of G includes a minimum-weight edge among those with one endpoint in each subset.

   (b) Prove that for any cycle in G, the minimum spanning tree of G excludes the maximum-weight edge in that cycle.

4. A **polygonal path** is a sequence of line segments joined end-to-end; the endpoints of these line segments are called the **vertices** of the path. The **length** of a polygonal path is the sum of the lengths of its segments. A polygonal path with vertices $(x_1, y_1), (x_2, y_2), ..., (x_k, y_k)$ is **monotonically increasing** if $x_i < x_{i+1}$ and $y_i < y_{i+1}$ for every index i – informally, each vertex of the path is above and to the right of its predecessor.

   Suppose you are given a set S of n points in the plane, represented as two arrays $X[1...n]$ and $Y[1...n]$. Describe and analyze an algorithm to compute the length of the maximum-length monotonically increasing path with vertices in S.

5. Suppose we are given an n-digit integer X. Repeatedly remove one digit from either end of X (your choice) until no digits are left. The *square-depth* of X is the maximum number of perfect squares that you can see during this process.

   Describe and analyze an algorithm to compute the square-depth of a given integer X, represented as an array $X[1...n]$ of n decimal digits. Assume you have access to a subroutine ISSQUARE that determines whether a given k-digit number is a perfect square in $O(k^2)$ time.

6. Consider a full binary tree of height h. Starting from the root, flip a coin and go the left subtree with probability 1/2 (if you get a head), and to the right subtree with

probability 1/2 (if you get a tail). Continue to flip coins and take subtrees until you arrive at a leaf.

Define the random variable

$$X_h = \alpha^k,$$

where k is the number of left subtrees taken on the walk down to the leaf. Prove that $E[X_h] = (\frac{1+\alpha}{2})^h$.