TOBB ETÜ, BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BİL 331/531 Design and Analysis of Algorithms

HOMEWORK 4 (37 Points)

Due Date: February 27, 2018

1 [6 Points] Finding Intersection of Two Sets

Let $K = \{1, 2, \ldots, k\}$ be the set of k smallest positive integers. Given two subsets $A, B \subset K$ of K, we want to efficiently find $A \cap B$.

Precisely, you are asked to give an $O(k)$-algorithm that takes two linked list A and B as parameters, and returns a linked list C. The elements of A are a subset of K, and the same holds for the linked list B. The elements of C should be the intersection of elements of A and B.

2 [4 Points] Odds are Even

Prove that the number of nodes of odd degree in an undirected graph is always even.

3 [4 Points] Number of Friends at a Party

Prove that, at any party with at least two people, there must be two individuals who know the same number of people present at the party. (It is assumed that the relation *a knows b* is symmetric.)
(Hint: You should have learned about the Pigeonhole Principle in BIL 132. If you forgot what it is, it is time to relearn.)

4 [6 Points] Understanding Search Trees

We have a connected graph $G = (V, E)$, and a specific vertex $u \in V$. Suppose we compute a depth-first search tree rooted at u, and obtain a tree T that includes all nodes of G. Suppose we then compute a breadth-first tree rooted at u, and obtain the same tree T. Prove that $G = T$. (In other words, if T is both a depth-first search tree, and a breadth-first search tree rooted at u, then G cannot contain any edges that do not belong to T.)

5 [7 Points] An Application in Wireless Networks

Assume you are working on a project for your Wireless Networks class. You are currently studying the properties of a network of n mobile devices. As the devices move around, they define a graph at any point in time as follows: there is a node representing each of the n devices, and there is an edge between device i and device j if the physical locations of i and j are no more than 500 meters apart (If so, we say that i and j are *in range* of each other). You'd like it to be the case that the network of devices is *connected* at all times, and so you have constrained the motion of the devices to satisfy the following property: at all times, each device i is within 500 meters of at least $\frac{n}{2}$ other devices. (We will assume that n is even). What would you like to know is: Does this property by itself guarantee that the network will always remain connected?

Here is a concrete way to formulate the question as a claim about graphs.

Claim: *Let G be a graph on n nodes, where n is an even number. If every node of G has degree at least $\frac{n}{2}$ (i.e., connected to at least $\frac{n}{2}$* ***other*** *nodes), then G is connected.*

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

6 [10 Points] An Application in Robotics

You are working on techniques for coordinating groups of mobile robots. Each robot has a radio transmitter that it uses to communicate with a base station, and if robots gets too close to one another, then there are problems with interference among the transmitters. So a natural problem arises: how to plan the motion of the robots in such a way that each robot gets to its intended destination, but in the process the robots don't come close enough together to cause interference problems.

You can model this problem as follows. Suppose that we have an undirected graph $G = (V, E)$, representing the floor plan of the building, and there are two robots initially located at nodes a and b in the graph. The robot at node a wants to travel to node c along a path in G, and the robot at node b wants to travel to node d. This is accomplished by means of a *schedule*: at each time step, the schedule specifies that **exactly one** of the robots move across a single edge, from one node to a neighboring node; at the end of the schedule, the robot from node a should be sitting on c, and the robot from b should be sitting on d.

A schedule is *interference-free* if there is no point at which the two robots occupy nodes that are at a distance $\le r$ from one another in the graph, for a given parameter r. We'll assume that the two starting points a and b are at a distance greater than r, and so are the two ending nodes c and d.

[5 Points] Give a polynomial-time algorithm that decides whether there exists an interference-free schedule by which each robot can get its destination.

[5 Points] If your algorithm works in time $O(n^3)$, where $n = |V|$, then you get additional 5 points.

(Hint: A configuration for robots is an *ordered pair* (u, v), denoting the situation where the first robot is at node u, and the second node is at node v. Create a new graph H, where there is a corresponding node for every configuration (u, v) such that the distance between u and v is greater than r. Notice that you need to join two nodes of H by an edge if they represent configurations that could be consecutive in a schedule. You should be able to solve this problem, after this much hint.)