CSE 3101: Design and Analysis of Algorithms

Assignment 5, Weight: 7%, Due: Aug 2, in the drop box by 5 pm

1. (2 points) Question 23-4, parts b and c only. This is on page 641 in the third edition.

2. (3 points) A war is being fought between two countries, A and B. As a loyal citizen of country C, you decide to help your country's espionage by attending the peace-talks taking place these days (incognito, of course). There are n people at the talks (not including you), but you do not know which person belongs to which country. You can see people talking to each other, and by observing their behaviour during their occasional one-to-one conversations, you can guess if they are friends or enemies. In fact what your country would need to know is whether certain pairs of people are from the same country, or they are enemies. You may receive such questions from C's government even during the peace-talks, and you have to give replies on the basis of your observations so far. Fortunately nobody talks to you, as nobody pays attention to your humble appearance.

   Friendship (denoted $\diamond$) is an equivalence relation, i.e.

   - If $x \diamond y$ and $y \diamond z$ then $x \diamond z$ (The friends of my friends are my friends as well).
   - If $x \diamond y$ then $y \diamond x$ (Friendship is mutual).
   - $x \diamond x$ (Everyone is a friend of himself).

   The enemy relationship (denoted *) is symmetric and irreflexive

   - If $x * y$ then $y * x$ (Hatred is mutual).
   - Not $x * x$ (Nobody is an enemy of himself).

   Also

   - If $x * y$ and $y * z$ then $x \diamond z$ (A common enemy makes two people friends.)
   - If $x \diamond y$ and $y * z$ then $x * z$ (An enemy of a friend is an enemy.)

   After a period of observation, you record a list of pairs of people that are friends and a list of pairs of people that are enemies. Assume that this information is consistent (i.e., does not contradict itself). Describe an algorithm that uses this information in answering queries like "are (x, y) friends?", "are (x, y) enemies?". Provide a formal proof of correctness, or argue informally that your algorithm is correct. Analyze the algorithm described.

   Suppose you are not sure that your observations are consistent (there are double agents?) and wish to detect inconsistencies in the information given. How can you modify your algorithm to do this?

3. (2 points) Suppose you are organizing a summer sports camp. The camp has at least one counselor who is skilled at each of the n sports covered by the camp (there need not be one counselor qualified in all n sports). You receive applications from m potential counselors. For each of the n sports there is some subset of the m applicants qualified in that sport. For a given number $k < m$, is it possible to hire at most k of the counselors and have at least one counselor qualified in each of the sports? Show that this problem is NP-complete.