CSE 450/598 Design and Analysis of Algorithms
Quiz II, Spring'12
Closed Books, Closed Notes
Time: 45 minutes
Answer any two questions

Problem 1: Suppose that the capacity of a cut $(S : \overline{S})$ in a network, can be defined in the following two ways:

$$(i)\ C(S:\overline{S}) = \sum_{e \in (S:\overline{S})} C(e), \qquad (ii)\ C(S:\overline{S}) = \mathtt{min}\Big(\sum_{e \in (S:\overline{S})} C(e), \sum_{e \in (\overline{S}:S)} C(e)\Big)$$

where $C(e)$ represents the capacity of the edge e. Will *maximum flow* from a source node s to a destination node t be equal to *capacity of the minimum cut* in case definition (i) is used? Will it still be true if definition (ii) is used? If your answer "yes" then explain why this will be true. If your answer "no" then provide a counterexample.

Using Ford-Fulkerson algorithm compute the maximum flow from node Vancouver to Winnipeg in the network shown in Fig 1. Find the minimum cut in this network. Show all your work.

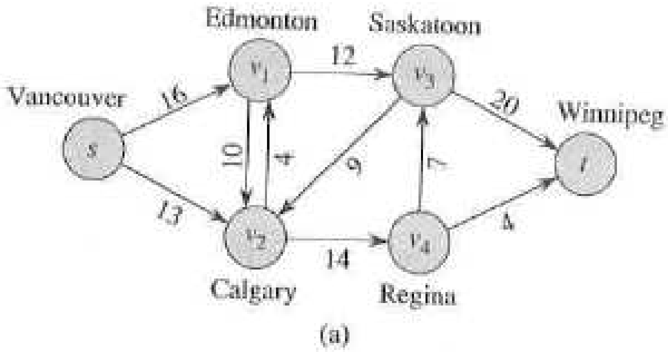

Figure 1: Flow Network

Problem 2: The activity selection problem (ASP) is described as follows: Suppose we have a set $A = \{a_1, \ldots, a_n\}$ of n proposed activities that wish to use a resource, such as lecture hall, which can be used by only one activity at a time. Each activity i has a start time s_i and a finish time f_i, where $s_i \leq f_i$, . If selected, activity i takes place during the half-open time interval $[s_i,\ f_i)$. Activities i and j are compatible if the intervals $[s_i,\ f_i)$ and $[s_j,\ f_j)$ do not overlap (i.e., i and j are compatible if $s_i \geq f_j$ or $s_j \geq f_i$. In one version of the activity selection problem the objective is to select a largest set of mutually compatible activities. In a second version of the problem the objective is to find the smallest number of resources (lecture halls) needed to schedule *all* the activities.

Suppose the following algorithm is used to solve the second version of the ASP problem. The algorithm loops between two phases until all the activities are scheduled. In phase I, a set of compatible activities are identified using the Algorithm_Activity_Selection (AAS). In phase II, this set of compatible activities is assigned to a resource (e.g., a lecture hall). The algorithm removes this set of activities from further consideration and among the remaining set of activities, finds another set of compatible activities using AAS. The process continues till all the activities are assigned to a resource.

For AAS, assume that the activities are ordered by monotonically increasing finish times $f_1 \leq f_2 \leq f_3 \leq \ldots \leq f_n$. If not, we can sort them into this order in time O(n lg n), breaking ties arbitrarily.

Algorithm_Activity_Selection (AAS)
begin
    $n \leftarrow |A|$ where $A = \{a_1, \ldots, a_n\}$;
    $CA \leftarrow \{a_1\}$;
    $i \leftarrow 1$;
    for $j := 2$ **to** n **do**
        if $s_j \geq f_i$ **then**
            begin
                $CA \leftarrow CA \cup \{a_j\}$
                $i \leftarrow j$;
            end
    return CA
end

Algorithm_Resource_Allocation
begin
    $A = \{a_1, \ldots, a_n\}$; (A is the set of activities)
    $R := 0$; (R is the number of resources used)
    do until $(A = \emptyset)$
        begin
            Using Algorithm Activity Selection, find a set of compatible activities CA;
            Assign a resource to this set of compatible activities and increment R ($R := R + 1$);
            $A := A - CA$
        end
    return R
end

Question: Is this algorithm always going to find a smallest number of resources in which all the activities can be scheduled? If your answer is "yes" then prove it by providing arguments in support of your answer. If your answer is "no" then provide an instance of the problem, where the algorithm fails to find a smallest number of resources in which all the activities can be scheduled.

Problem 3: Define an NP-Complete problem. Prove that the 5-SAT problem NP-Complete. Show all your work.