Mid-semester examination: CS31005
Department of Computer Science and Engineering
IIT Kharagpur

Algorithms-II: LTP 3-1-0: Credits 4: Instructor: S P Pal: Time 2 hours: Marks 100
Autumn 2016

1. Consider the problem of computing the *minimum cardinality vertex cover* in a bipartite graph $G(A \cup B, E)$ where (i) all the edges in the edge set E have one vetex in A and another vertex in B, (ii) we are given a *maximum cardinality matching* M in G.

   Suppose we select the vertex with larger degree in each edge of M in the proposed minimum cardinality vertex cover; if both vertices in an edge of M have the same degree then we choose any arbitrary one of them in the proposed vertex cover. Will this method give us a minimum cardinality vertex cover? [The degree of a vertex is the number of edges in G incident on the vertex.]

   Justify your answer.

   [10 marks]

2. The *next fit* 0-1 bin-packing heuristic fills bins one by one with the given set of items by checking whether the last filled bin can accomodate the current item being considered. If the last bin cannot accomodate the current item then the current item uses a new bin. All bins are of capacity one and each item has size at most one. Show that the number of bins used by this heuristic is at most twice the minimum number of bins required for the given set of items.

   [10 marks]

3. Consider the greedy algorithm for computing a weighted set cover, where positive weights are assigned to the sets. There are n elements and $n+1$ sets, where the ith element e_i has weight $\frac{1}{n-i+1}$. The first n sets are $S_i = \{e_i\}$ where e_i is the ith element, $i \leq i \leq n$. The last set is $S_{n+1} = \{e_1, e_2, ..., e_n\}$ and has weight $1 + \epsilon$, where $0 \leq \epsilon \leq 1$. Here, *epsilon* is a constant with respect to n.

   What is the weight of the set cover computed by the greedy algorithm? Is this the minimum weight for a set cover in the above system? For what values of n with respect to ϵ is the weight of the minimum weighted set cover $1 + \epsilon$?

   Explain your answers justifying them.

   [8+4+3 marks]

4. We wish to minimize $\sum_{i=1}^{6} x_i$, where $x_1, x_2, x_3, x_4, x_5, x_6$ are all non-negative integers and the following seven inequalities are satisfied. Namely, $x_1+x_4 \leq 1$, $x_1+x_5 \leq 1$, $x_1+x_6 \leq 1$, $x_2+x_4 \leq 1$, $x_2+x_5 \leq 1$, $x_3+x_5 \leq 1$, and $x_2+x_6 \leq 1$. Determine the minimum value sought above. Also, derive the dual linear program for the relaxed version of the

above integer linear program. Determine the maximum objective function value for feasible solutions of this dual relaxation.

Justify and explain your answers.

[12+8 marks]

5. Precisely state the NP-complete decision version problem statement for the vertex cover problem for undirected graphs. State the decision version statement for the independent set problem for graphs. Using the NP-compleness of the vertex cover problem above, show that the decision version of your independent set problem too is NP-complete.

   [5+5+5 marks]

6. Given an integer $n > 10$ and a positive integer r, $1 \leq r < n$, how would you verify that r is indeed a primitive root of the prime number n? Is your procedure efficient? Explain. Illustrate for $n = 23$ for a suitable primitive root r of 23.

   [5+5+5 marks]

7. Show that a network with integral capacities for all edges has a maximum flow function $f_{s,t}$ for source s and sink t (s and t are vertices in the network), such that $f_{s,t}$ assigns integral amounts of flow to each edge of the network.

   [7 marks]

8. State the *max-flow-min-cut* theorem. Define the term (s,t)-*cut capacity* of an (s,t)-*cut* in a *network* $G(V, E, c)$, where s and t are the *source* and the *sink* vertices and c is the *capacity function* that assigns a positive *capacity* $c(e)$ to each directed edge e of the network G.

   Prove that the *maximum flow* from s to t in G is at most the capacity of each (s,t)-cut in G.

   [3+2+3 marks]

==