Norwegian University of Science and Technology
Department of Computer and Information Science

Final Exam in
TDT4125 Algorithm Construction, advanced course

Saturday May 24, 2014
Time: 09:00–13:00

Aids: (B) All printed/handwritten; specific, simple calculator

Language: English

Please read the entire exam before you start, plan your time, and prepare any questions for when the teacher comes to the exam room. Make assumptions where necessary. Keep your answers short and concise. Long explanations that do not directly answer the questions are given no weight.

Each subtask counts equally toward the final score.

Problem 1

a) In an approximation technique, we construct solutions by iteratively increasing variables that represent violated constraints. What is this technique called?

b) A relation in the curriculum expresses the relationship between the average value for non-negative random variables and the probability of high values. What is the name of this relation?

c) Assume that you have a problem with an exponential number of constraints. Describe briefly how you would proceed to solve the problem in polynomial time, and what is required for your method to succeed.

d) Describe briefly and precisely how you would turn a randomized approximation algorithm into a deterministic one. Which property is it one desires from the resulting algorithm, and how is this achieved?

Problem 2

a) From a group A of n people, you are to select a group B of k representatives for a board. For each pair $p = \{x, y\} \subseteq A$ of people you have estimated a value $w(p)$ for how much they agree, and thereby how well they could represent each other's opinions. You wish the selection of representatives to have a maximum sum of these values. That is, you wish to maximize

$$w(B) = \sum_{p \in (A \setminus B) \times B} w(p) .$$

You would even be satisfied with an algorithm that would give you an answer that was within a constant factor of the optimum, but after a while you realize that even this would be problematic. Show why this is so.

Problem 3 Figure 1 shows a partitioning of a set S in the hierarchical cut decomposition. The set S is defined by the large, complete circle; subdivisions are indicated with small, dashed circles.

a) Write the subsets that S will be partitioned into given each of the following three random permutations:

$$\pi_1 = 1, 2, 3, 4, 5, 6$$
$$\pi_2 = 6, 5, 4, 3, 2, 1$$
$$\pi_3 = 3, 2, 1, 4, 5, 6$$

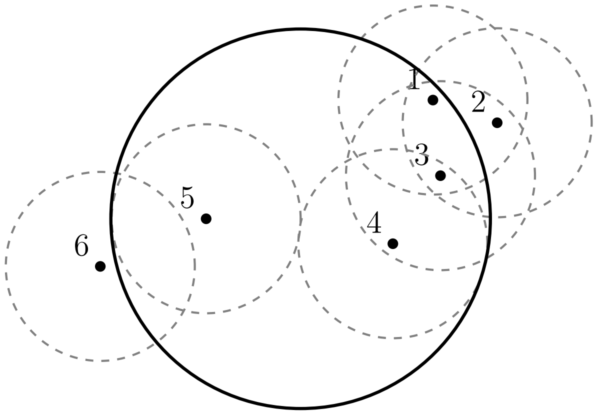

Figure 1: Hierarchical cut decomposition of a set S.

b) Given the random permutation π_3 from a) and the resulting subsets, what is the size of the largest subset determined in the next iteration of the hierarchical cut decomposition? Explain your answer.

Problem 4 Give polynomial-time algorithms for the following:

a) Coloring a 2-colorable graph with two colors.

b) Coloring a graph of maximum degree Δ with $\Delta + 1$ colors.

Problem 5 In the undirected maximum cut problem, we are given as input an undirected graph $G = (V, E)$ and a non-negative weight w_{ij} for each edge $\{i, j\} \in E$. The goal is to partition V into two sets U and $W = V - U$ so as to maximize the total weight of the edges going between U and W (that is, edges $\{i, j\}$ with $i \in U$ and $j \in W$). Consider the following algorithm for solving this problem: Number the nodes $1 \dots n$. In the first iteration, add node 1 to U. In iteration k, place node k in either U or W. To decide which, consider all edges $F = \{\{k, j\} \in E : 1 \leq j \leq k - 1\}$. We choose to place node k in U or W based on what will maximize the number of edges in F.

a) Show that this is a 1/2-approximation.

In the maximum directed cut problem, we are given as input a directed graph $G = (V, A)$, with non-negative weight w_{ij} for each arc $(i, j) \in A$. The goal is to partition V into two sets U and $W = V - U$ so as to maximize the total weight of the arcs going from U to W (that is, arcs (i, j) with $i \in U$ and $j \in W$).

b) Show that this problem can be expressed as an integer quadratic program. (Hint: it may help to introduce a variable y_0 that indicates whether the value -1 or 1 means that y_i is in the set U.).

Problem 6

a) Prove that the following shortest s–t path algorithm is equivalent to Dijkstra's algorithm:

PRIMAL–DUAL SHORTEST s–t PATH ALGORITHM (ALG. 7.4):

$y \leftarrow 0$
$F \leftarrow \emptyset$
while there is no s–t path in (V, F)
    Let C be the connected component of (V, F) containing s
    Increase y_C until there is an edge $e' \in \delta(C)$ such that $\Sigma_{S \in \mathcal{S}: e' \in \delta(S)} y_S = c_{e'}$
    $F \leftarrow F \cup \{e'\}$
end
Let P be an s–t path in (V, F)
return P

Problem 7 Consider the problem of finding the shortest tour of a complete (undirected) graph. Each edge $\{i, j\}$ has a positive (non-zero) weight of w_{ij}. Now consider the following two scenarios, for some constant $k > 0$.

Scenario 1: For every sequence of nodes $x, y, u, \ldots, v, z$, the following inequality holds:

$$w_{x,z} \leq k \cdot (w_{xy} + w_{yu} + \cdots + w_{vz})$$

Scenario 2: For every sequence of nodes $x, y, u, \ldots, v, z$, the following inequality holds:

$$w_{x,z} \leq k + (w_{xy} + w_{yu} + \cdots + w_{vz})$$

a) Briefly discuss the difference between the scenarios in terms of how well you could approximate the optimal solution.