Assignment 5

Algorithm Design and Analysis

November 28, 2016

Notice:

1. **Due** 9:00 a.m., Dec. 09 for hard copy and 11:55 p.m., Dec. 09 for digital version;
2. Please submit your answers in hard copy **AND** submit a digital version to UCAS website https://www2.ucas.ac.cn/ .
3. Please choose **at least 4** problems from Problem 1-7 and **at least 2** problems from Problem 8-10.
4. When you're asked to give an algorithm, you should do at least the following things:
   - Describe the basic idea of your algorithm in natural language **AND** pseudo-code;
   - Prove the correctness of your algorithm.
   - Analyse the complexity of your algorithm.

1 Load balance

You have some different computers and jobs. For each job, it can only be done on one of two specified computers. The load of a computer is the number of jobs which have been done on the computer. Give the number of jobs and two computer ID for each job. You task is to minimize the max load.

(hint: binary search)

2 Matrix

For a matrix filled with 0 and 1, you know the sum of every row and column. You are asked to give such a matrix which satisfys the conditions.

3 Unique Cut

Let $G = (V, E)$ be a directed graph, with source $s \in V$, sink $t \in V$, and nonnegative edge capacities c_e. Give a polynomial-time algorithm to decide whether G has a unique minimum st cut.

4 Problem Reduction

There is a matrix with numbers which means the cost when you walk through this point. you are asked to walk through the matrix from the top left point to the right bottom point and then return to the top left point with the minimal cost. Note that when you walk from the top to the bottom you can just walk to the right or bottom point and when you return, you can just walk to the top or left point. And each point CAN NOT be walked through more than once.

5 Network Cost

For a network, there is one source and one sink. Every edge is directed and has two value c and a. c means the maximum flow of the adge. a is a coefficient number which means that if the flow of the edge is x, the cost is ax^2.

Design an algorithm to get the Minimum Cost Maximum Flow.

6 Maximum Cohesiveness

Given an undirected graph, each edge is assigned one weight, find a subset S of nodes to maximize $e(S)/|S|$, where $e(S)$ denotes the sum of edge weights in S and $|S|$ is the number of nodes in S. Give a polynomial-time algorithm that takes a rational number α and de- termines whether there exists a set S with cohesiveness at least α.

7 Maximum flow

Another way to formulate the maximum-flow problem as a linear program is via flow decomposition. Suppose we consider all (exponentially many) s-t paths p in the network G, and let f_p be the amount of flow on path p. Then maximum flow says to find

$$\begin{array}{lrcl} \max & z & = & \sum f_p \\ s.t. & \sum_{e \in p} f_p & \leq & u_e, \quad for \quad all \quad edge \quad e \\ & f_p & \geq & 0 \end{array}$$

(The first constraint says that the total flow on all paths through e must be less than ue.) Take the dual of this linear program and give an English explanation of the objective and constraints.

8 Ford-Fulkerson algorithm

Implement Ford-Fulkerson algorithm to find the maximum flow of the following network, and list your intermediate steps. Use you implementation to solve problem 1 and show your answers.
INPUT: (N, M) means number of jobs and computers. Next N line, each line has two computer ID for a job. see more detial in the file problem1.data.
OUTPUT: the minimum number of the max load.

9 Push-relabel

Implement push-relabel algorithm to find the maximum flow of a network, and list your intermediate steps. Use your implementation to solve problem 2 and write a check problem to see if your answer is right.
INPUT: Numbers of rows and columns. And the sum of them. See more detial in the file problem2.data.
OUTPUT: The matrix you get. Any one satisfy the conditions will be accept.

10 Cycle canceling

Implement Cycle canceling algorithm to find the minimum cost flow of a network, and list your intermediate steps.
INPUT: A directed graph $G =< V, E >$. Each edge e has a capacity c_e and a cost w_e. Two special points: source s and sink t. Please make the input in DIMACS format. For DIMACS format, see DIMACS maximum flow problems.html in this folder.
OUTPUT: For each edge e, to assign a flow f_e such that $\sum_{e \in E} f_e w_e$ is minimized.