Advanced Algorithms

Problem set #4
Due date: 20 DEY 98 (via lms.iut.ac.ir)

Solutions to the problems on this set might be found on the Internet. It is not acceptable to copy such solutions. Write with your own notation and formulation.

Problem 1. Answer the following parts,

(a) Show that the dual of a linear program is the original program itself.

(b) Show that any minimization program can be transformed into an equivalent program in standard form

(c) Convert the following LP into standard form:

$$\begin{gathered} \min\ 2x_1 + 7x_2 + x_3 \\ \textbf{s.t}\ x_1 - x_3 = 7 \\ 3x_1 + x_2 \geq 24 \\ x_2 \geq 0 \\ x_3 \leq 0. \end{gathered}$$

(d) Convert the following LP into slack form:

$$\begin{gathered} \max\ 2x_1 - 6x_3 \\ \textbf{s.t}\ x_1 + x_2 - x_3 \leq 7 \\ 3x_1 - x_2 \geq 8 \\ -x_1 + 2x_2 + 2x_3 \geq 0 \\ x_1, x_2, x_3 \geq 0. \end{gathered}$$

Problem 2. Present a geometric interpretation (Solve the problem graphically) for the following LP:

$$\begin{gathered} \max\ z = x_1 + 5x_2 \\ \textbf{s.t}\ 2x_1 + 4x_2 \geq 14 \\ x_1 + x_2 \geq 7 \\ -x_1 + 3x_2 \leq 5 \\ 6x_1 - x_2 \geq 14 \\ x_1, x_2 \geq 0. \end{gathered}$$

Problem 3. Solving with two-phase simplex method:

(a) Convert the following LP into standard form:

$$
\begin{gathered}
\max\ 2x_1 + x_2 \\
\textbf{s.t}\ 2x_1 + x_2 \le 6 \\
2x_1 + 3x_2 \le 5 \\
4x_1 + x_2 \le 6 \\
x_1 + 5x_2 \le 2 \\
x_1, x_2 \ge 0.
\end{gathered}
$$

(b) Convert the following LP into slack form:

$$
\begin{gathered}
\max\ 2x_1 - 6x_2 \\
\textbf{s.t}\ -x_1 - x_2 - x_3 \le -2 \\
2x_1 - x_2 + x_3 \le 1 \\
x_1, x_2, x_3 \ge 0.
\end{gathered}
$$

Problem 4. Solving the following LP using Bland's rule to resolve degeneracy.

$$
\begin{gathered}
\max\ 10x_1 - 57x_2 - 9x_3 - 24x_4 \\
\textbf{s.t}\ 0.5x_1 - 5.5x_2 - 2.5x_3 + 9x_4 \le 0 \\
0.5x_1 - 1.5x_2 - 0.5x_3 + x_4 \le 0 \\
x_1 \le 1 \\
x_1, x_2, x_3, x_4 \ge 0.
\end{gathered}
$$

Problem 5. Let $G = (V, E)$ be an undirected graph, with weights w_e on edges. The following is an exact LP-relaxation for the problem of finding a maximum weight matching in G.

$$
\begin{aligned}
&\max \sum_e w_e x_e \\
\textbf{s.t} \quad &\sum_{e:\ incident\ at\ v} x_e \le 1, \qquad v \in V \\
&\sum_{e:\ e \in S} x_e \le \frac{|S| - 1}{2}, \qquad S \subset V, |S|\ odd \\
&x_e \ge 0, \qquad e \in E.
\end{aligned}
$$

Obtain the dual of this LP.

Problem 6(Extra Credit). Given a sequence of positive numbers $x_1, \ldots, x_n$ and an integer k, design a polynomial time algorithm that outputs

$$
\sum_{S \in \binom{n}{k}} \prod_{i \in S} x_i,
$$

where the sum is over all subsets of size k.

Problem 7(Extra Credit). The spanning-tree game is a 2-player game. Each player, in turn, selects an edge. Player 1 starts by deleting an edge, and then player 2 fixes an edge (which has not been deleted yet); an edge fixed cannot be deleted later on by the other player. Player 2 wins if he succeeds in constructing a spanning tree of the graph; otherwise, player 1 wins. The question is which graphs admit a winning strategy for player 1 (no matter what the other player does), and which admit a winning strategy for player 2. Show that player 1 has a winning strategy if and only if G does not have two edge-disjoint spanning trees. Otherwise, player 2 has a winning strategy.

Problem 8(Extra Credit). Consider an infinite sequence of positions $1, 2, 3, \ldots$ and suppose we have a pebble at position 1 and another pebble at position 2. In each step, we choose one of the pebbles and move it according to the following rule: Say we decide to move the pebble at position i; if the other pebble is not at any of the positions $i+1, i+2, \ldots, 2i$, then it goes to $2i$, otherwise it goes to $2i+1$. For example, in the first step, if we move the pebble at position 1, it will go to 3 and if we move the pebble at position 2 it will go to 4. Note that, no matter how we move the pebbles, they will never be at the same position. Use induction to prove that, for any given positive integer n, it is possible to move one of the pebbles to position n. For example, if $n = 7$ first we move the pebble at positions 1 to 3. Then, we move the pebble at positions 2 to 5 Finally, we move the pebble at positions 3 to 7.

Good Luck and Merry Christmas.