AMS 550.666: Combinatorial Optimization
Homework Problems - Week II

1. We are given a general digraph $G$ with edge capacities $u_e \geq 0$. Also, for every node $v \in V$, we have a real number $b_v$ (not necessarily nonnegative).

   i. Give an efficient algorithm to solve the following general version of the transshipment problem discussed in class. Find a set of real numbers $x_e$ for each edge $e \in E$ such that $f_x(v) = b_v$ for all $v \in V$, and $0 \leq x_e \leq u_e$. This is called a generalized flow. In class, we looked at a special case where all edges are between a vertex with $b_v \leq 0$ and a vertex with $b_v \geq 0$ (the factory-customer problem). Now we are considering the problem on a more general digraph.

   ii. For any subset $S \subseteq V$, let $b(S) = \sum_{v \in S} b_v$. Prove that there is a feasible solution to the generalized problem if and only if $b(V) = 0$ and $b(S) \leq u(\delta(V \setminus S))$ for every nonempty $S \subseteq V$.

2. (Open Pit Mining) In the design of an open pit mine, the region under consideration is divided into 3D blocks. Then decisions are made as to which blocks will be excavated. The only constraint is that if a block needs to be excavated, then any block “above” it also needs to be excavated (with a suitable definition for “above”). This leads to the following abstraction: Consider a digraph $G = (V, E)$ where each node had an associated weight (benefit) $b_v \in \mathbb{R}$ (not necessarily nonnegative). We want to find a subset of nodes to maximize our benefit. However, if choose a node $v$ and there is an edge $vw \in E$, then $w$ must also be chosen. So the goal is to find a subset of vertices $S \subseteq V$ such that $\delta(S) = \emptyset$ and the total benefit $b(S)$ is maximized.

   Give an efficient algorithm to solve this problem.

3. (Problem 3.16 from textbook) Find the maximum matching and minimum cover in the following bipartite graph.

4. (Problem 3.17 from textbook) We are given a bipartite graph $G = (V, E)$ with nonnegative weights $b_v$ on vertices $v \in V$. Give an algorithm to find the minimum weight vertex cover for $G$.

5. (Problem 3.18 from textbook) Let $G = (V = V_1 \uplus V_2, E)$ be a bipartite graph and a nonnegative integer $b_v$ associated with each vertex $v \in V$. Find a polynomial time algorithm to decide if there exists a subset of edges $E' \subseteq E$ such that for every vertex $v$, there are exactly $d_v$ edges incident on it.

6. (Problem 3.22 from textbook) We have a family $(S_1, \ldots, S_k)$ of subsets of a set $Q$. A system of distinct representatives (SDR) is a set of distinct elements $\{q_1, \ldots, q_k\} \subseteq Q$ such that $q_i \in S_i$ for all $i = 1, \ldots, k$. Prove that a family has an SDR if and only if every subset $I$ of $\{1, \ldots, k\}$ we have $|\bigcup(S_i : i \in I)| \geq |I|$.
7. (Problem 3.23 from textbook) Suppose every node of a bipartite graph has degree \( p \geq 1 \). Show that there exists a matching that covers every vertex. Such a matching is called a perfect matching. Deduce that the graph has \( p \) disjoint perfect matchings (meaning no two matchings share a common edge). [Hint: Problem 6 could be useful.]

8. (Problem 3.26 from textbook) For families \((S_1, \ldots, S_k)\) and \((T_1, \ldots, T_k)\) of subsets of \(Q\), a common SDR is a set \( \{q_1, \ldots, q_k\} \subseteq Q \) that is an SDR for both families. Prove that there exists a common SDR if and only for every pair \(I, J\) of subsets of \(\{1, \ldots, k\}\) we have

\[
|\left(\bigcup\{S_i : i \in I\}\right) \cap \left(\bigcup\{T_j : j \in J\}\right)| \geq |I| + |J| - k.
\]

[Hint: Set up a max-flow model to solve the common SDR problem, and use the max flow-min cut theorem]

9. (Problem 3.29 from textbook) Projects \(P_1, P_2, \ldots, P_k\) are available to be undertaken, and there is a set of resources \(R_1, \ldots, R_\ell\) that can be used to complete these projects. With each project \(i\) that is undertaken, we get a nonnegative revenue \(r_i\). Each project \(i\) requires a set \(S_i \subseteq \{R_1, \ldots, R_\ell\}\) of resources to be available, and each resource \(R_j\) has an associated cost \(c_j\). However, if resource \(R_j\) is purchased, it is available for any project for which it is required. Give an algorithm to choose a set of projects (and the required set of resources to buy) to maximize the revenue minus the cost.