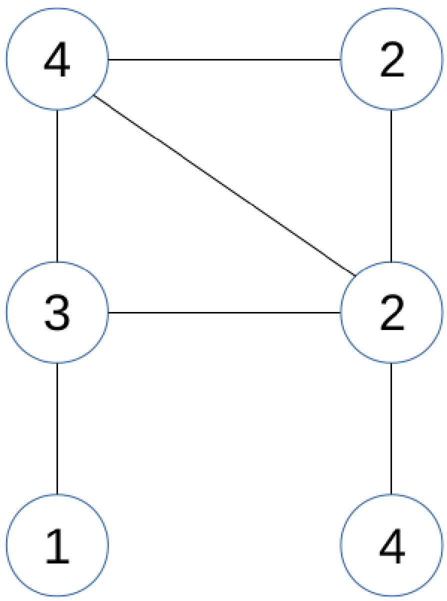
Homework 2
Integer LP and Relaxation

1. (5 points) In the previous problem set, we solved the transportation problem of the space colonies using LP. Explain why the solution was integral with a proof.

2. Recall the Minimum Weight Vertex Cover (VC) Problem: Given an undirected graph $G = (V, E)$, with node set V and edge set E, where each node has a positive weight $w(v)$ associated with it (see figure), the goal is to select a subset $V' \subseteq V$ of nodes such that every edge has at least one node incident to it, and the total selected node weight $\sum_{v \in V'} w(v)$ is minimized.
   (a) (4 points) Formulate the ILP for the VC problem for the example below, and solve it using the integer solver `cvxopt.glpk.ilp()`
   (b) (2 points) Pass to the LP relaxation and solve it using CVXOPT and comment on the relation between the two solutions.
   (c) (2 points) Apply the rounding rule discussed in class to the optimal LP solution to obtain a solution to the ILP and compare it to the optimal ILP solution.



3. Consider a number of interpreters (Olof, Petra, Qamar, Rachel, Søren and Tao), as well as a set of languages (Arab, Bengali, Cantonese, Dutch, English, French and German). Each interpreter speaks a number of different languages (abbreviated by first letter), and has a certain per-diem integer cost:

Interpreter	Languages	Cost
O	ABD	3
P	C	1
Q	CDG	1
R	B	2
S	G	4
T	EF	1

(a) (2 points) A *hypergraph* is a structure $H = (V, E)$ where V is a set of vertices and E is a collection of subsets of V. The special case when all subsets $e \in E$ have size exactly 2 corresponds to the familiar case of a graph. A vertex cover in such a hypergraph is a subset $U \subseteq V$ such that $e \cap U \neq \emptyset$ for each $e \in E$ (note that this reduces to the usual vertex cover in graphs). Show that the problem of finding interpreters can be formulated as a vertex cover problem in a sutable hypergraph.

(b) (4 points) Develop a ILP formulation to finding the vertex cover of minimum cost in a hypergraph. The hypergraph can be represented as a $|V| \times |E|$ 0/1 matrix A where $A[i, j] = 1$ iff vertex i is in edge j and the costs for vertices are in an array `c` where the cost of picking vertex i is $c[i]$. Use the ILP formulation for the VC problem to hire the cheapest set of interpreters such that all languages are covered. Input the data above manuallly and solve it using `cvxopt.glpk.ilp()`.

(c) (2 points) Pass to the LP relaxation and solve it using CVXOPT.

(d) (2 points) Explain why the two solutions above are same (different).

4. Consider the ILP and its LP relaxation corresponding to the VC problem for the graph G given in the data file. This is a *random graph* $G(n, p)$ with $n = 200$ vertices generated as follows: for each pair of vertices *independently*, we add an edge with probability $p = 0.1$ (so the graph has about 2000 edges).

(a) (2 points) Find the optimal solution using `cvxopt.glpk.ilp()`.

(b) (2 points) Solve the LP relaxation using CVXOPT and apply the rounding rule discussed in class to obtain a vertex cover. Compare it to the optimal solution in part (a).

(c) (6 points) Consider the following rounding rule: we build up the vertex cover incrementally starting with $S := \emptyset$. Now consider the edges in G in any order. If an edge (u, v) is already covered by a vertex in S, do nothing. Otherwise add to S the vertex u if $x^*(u) \geq x^*(v)$, or v otherwise (where $\mathbf{x}^*$ is the LP optimum solution computed in part (b). Comment why this also results in a vertex cover and has cost no more than that corresponding to the rounding rule in part (b). Compare the cost of the solution produced by this rule to the optimal solution.