Goals
Huffman’s coding, network flow, NP-completeness and reducibility

Questions
1. Huffman Coding:

   (a) Show the Huffman trie that results from the following distributions (frequencies in parentheses):
       colon (100), space (605), comma (705), 0 (431), 1 (242), 2 (176), 3 (59), 4 (185), 5 (250), 6
       (174). (To make the trie fully-specified, put the lower-weight subtree on the right on each merge
       operation. Start by listing the weights in increasing order from left to right, with their symbols
       below them on the next line, leaving space above to build trees.

   (b) What is the resulting binary code for the most frequent symbol? The least frequent?

   (c) With the coding scheme above, code the 7-symbol text “04: 12,”. Show the binary string and
       the bytes in hex.

   (d) With the coding scheme above, decode 011101001011001.

2. Max-Flow-Min-Cut: K&T 7.2: Given the following flow network on which an s-t flow has been
   computed. The capacity of each edge appears as a label on the edge, and the numbers in parentheses
   give the amount of flow sent on each edge. (Edges without parentheses – specifically, the four edges of
   capacity 3 – have no flow being sent on them.)

   (a) What is the value of this flow? Is this a maximum (s,t) flow in this graph?

   (b) Find a minimum s-t cut in the flow network and also say what its capacity is.

3. Max-Flow-Min-Cut: K&T 7.3: Given the following flow network on which an s-t flow has been
   computed. The capacity of each edge appears as a label on the edge, and the numbers in parentheses
   give the amount of flow sent on each edge. (as before, edges with no parentheses have no flow being
   sent on them.)
(a) What is the value of this flow? Is this a maximum (s,t) flow in this graph?
(b) Find a minimum s-t cut in the flow network and also say what its capacity is.

4. **Max-Flow-Min-Cut:** K&T 7.5: Decide whether the following statement is true or false. If it is true, give a short explanation. If it is false, give a counter example:

Given an arbitrary flow network, with a source s, a sink t, and a positive integer capacity $c_e$ on every edge $e$; and let $(A, B)$ be a minimum $s-t$ cut with respect to these capacities $\{c_e| e \in E\}$. Now suppose we add 1 to every capacity; then $(A, B)$ is still a minimum $s-t$ cut with respect to these new capacities $\{1 + c_e| e \in E\}$.

5. **NP-Completeness and Reducibility:** A clique in an undirected graph is a subset of vertices such that each pair of the vertices is joined by an edge in the graph. (Equivalently, a clique is a complete subgraph. You can think about it as a subset of the vertices s.t. everyone is friends with everyone else.) Given an undirected graph, we would like to find the largest clique. The corresponding decision problem is “Given a graph $G$ and a number $k$, does $G$ contain a clique of size $k$?”. Show that the maximal clique problem is NP complete by a reduction from the Independent set problem shown in class. **Hint:** Given a graph $G$, its complement graph $G^c$ is a graph with the same set of vertices such that for any pair of vertices $(u,v)$ there is an edge in $G$ if and there is not an edge in $G^c$. The rest of the reduction follows very easily. Don’t forget to show that the Clique is in NP and the reduction is polynomial (also very easy).

6. **NP-Completeness and Reducibility:** You want to be a celebrity chef by creating a new dish. You have a list of $n$ ingredients and you want to use as many of them as possible. However, some of them don’t go well with others. For every pair of ingredients $(i,j)$ there is a penalty score $p(i, j)$ – a real number which ranges from 0 if the two ingredients fit perfectly to 1 if they go really badly together. The input to the problem is the set of $n$ ingredients and a matrix $P$ of size $n \times n$ with $P[i][j] = p(i, j)$ (we only need the top half of it, actually). Consider the problem EXPERIMENTAL-CUISINE, in which you want to prepare a dish with at least $k$ ingredients out of the $n$ with a total penalty of at most $d$.

(a) Show that the problem is in NP (Remember, what I’m asking is to show that a solution can be verified polynomially, not that it can or cannot be obtained polynomially).
(b) Consider a reduction from INDEPENDENT-SET as follows: Given an instance to INDEPENDENT-SET, a graph $G = (V, E)$ with $n$ vertices and $m$ edges, construct an instance of the EXPERIMENTAL-CUISINE problem with $n$ ingredients such that $p(i, j) = 1$ if and only if vertices $(i, j)$ have an edge in $G$. Show this is a polynomial time and space reduction.
(c) Based on your answer to (b), show that EXPERIMENTAL-CUISINE is NP-complete by completing the “if and only if” part. Don’t miss any of the details. **(Hint:** The goal size of the set is $k$ and the goal $d$ is 0).