This is the last homework before Midterm 2.

1. Suppose we are given a two-dimensional array $A[1..m, 1..n]$ of non-negative real numbers. We would like to round $A$ to an integer matrix, by replacing each entry $x$ in $A$ with either $\lfloor x \rfloor$ or $\lceil x \rceil$, without changing the sum of entries in any row or column of $A$. For example:

$$
\begin{pmatrix}
1.2 & 3.4 & 2.4 \\
3.9 & 4.0 & 2.1 \\
7.9 & 1.6 & 0.5
\end{pmatrix}
\rightarrow
\begin{pmatrix}
1 & 4 & 2 \\
4 & 4 & 2 \\
8 & 1 & 1
\end{pmatrix}
$$

Describe and analyze an efficient algorithm that either rounds $A$ in this fashion, or reports correctly that no such rounding exists.

2. You’re organizing the Third Annual UIUC Computer Science 72-Hour Dance Exchange, to be held all day Friday, Saturday, and Sunday in Siebel Center.¹ Several 30-minute sets of music will be played during the event, and a large number of DJs have applied to perform. You need to hire DJs according to the following constraints.

- Exactly $k$ sets of music must be played each day, and thus $3k$ sets altogether.
- Each set must be played by a single DJ in a consistent musical genre (ambient, bubblegum, dancehall, horrorcore, trip-hop, Nashville country, Chicago blues, axé, laïkó, skiffle, shape note, Nitzhonot, J-pop, K-pop, C-pop, T-pop, 8-bit, Tesla coil, ...).
- Each genre must be played at most once per day.
- Each DJ has given you a list of genres they are willing to play.
- No DJ can play more than five sets during the entire event.

Suppose there are $n$ candidate DJs and $g$ different musical genres available. Describe and analyze an efficient algorithm that either assigns a DJ and a genre to each of the $3k$ sets, or correctly reports that no such assignment is possible.

3. Describe and analyze an algorithm to determine, given an undirected² graph $G = (V, E)$ and three vertices $u, v, w \in V$ as input, whether $G$ contains a simple path from $u$ to $w$ that passes through $v$.

¹Efforts to secure overflow space in ECEB were sadly unsuccessful.
²This adjective is important; if the input graph were directed, this problem would be NP-hard.