1. (a) Describe and analyze an efficient algorithm that either rounds A in this fashion, or correctly reports that no such rounding is possible.

Solution: Assume without loss of generality that every row and every column of A sums to an integer, because no legal rounding is possible otherwise. To simplify the problem, we write A as the sum of two $m \times n$ arrays I ("integer") and F ("fractional") by setting

$$I[i, j] := \lfloor A[i, j] \rfloor \quad \text{and} \quad F[i, j] = A[i, j] - I[i, j]$$

for each i and j. If F' is a legal rounding for F, then $I + F'$ is a legal rounding for A. So we just have to compute a legal rounding of the fractional matrix F. For example:

$$\begin{bmatrix} 1.2 & 3.4 & 2.4 \\ 3.9 & 4.0 & 2.1 \\ 7.9 & 1.6 & 0.5 \end{bmatrix} = \begin{bmatrix} 1 & 3 & 2 \\ 3 & 4 & 2 \\ 7 & 1 & 0 \end{bmatrix} + \begin{bmatrix} \mathbf{.2} & \mathbf{.4} & \mathbf{.4} \\ \mathbf{.9} & \mathbf{.0} & \mathbf{.1} \\ \mathbf{.9} & \mathbf{.6} & \mathbf{.5} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} 1 & 3 & 2 \\ 3 & 4 & 2 \\ 7 & 1 & 0 \end{bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{0} & \mathbf{1} \end{bmatrix} = \begin{bmatrix} 1 & 4 & 2 \\ 4 & 4 & 2 \\ 8 & 1 & 1 \end{bmatrix}$$

To find a legal rounding for the fractional matrix F, we construct a flow network G with the following vertices and edges:

- A source s, a vertex r_i for each row i, a vertex c_j for each column j, and a target t;
- An edge $s \to r_i$ with capacity $\sum_k F[i, k]$ for each row i;
- An edge $c_j \to t$ with capacity $\sum_k F[k, j]$ for each column j;
- An edge $r_i \to c_j$ with capacity 1 for each row i and column j such that $F[i, j] > 0$.

Next we compute a maximum flow f^* in G from s to t. Because every edge capacity is an integer, we can assume without loss of generality that f^* is an *integer* flow; in particular, every flow value $f^*(r_i \to c_j)$ is either 0 or 1. Finally, for each row i and column j, set $F^*[i, j] \leftarrow f^*(r_i \to c_j)$.

Now I claim that F^* is a legal rounding of F if and only if f^* saturates every edge leaving s (and therefore every edge entering t. There are two cases to consider:

- Suppose f^* saturates every edge leaving s and every edge entering t. Then for each row index i, we have

$$\begin{aligned} \sum_k F^*[i, k] &= \sum_k f^*(r_i \to c_j) && \text{[definition of } F^*[i, k]\text{]} \\ &= f^*(s \to r_i) && \text{[flow conservation at } r_i\text{]} \\ &= c(s \to r_i) && \text{[definition of saturated]} \\ &= \sum_k F[i, k] && \text{[definition of } c(s \to r_i)\text{]} \end{aligned}$$

  In short, every row of F^* has the same sum as the corresponding row of F. A symmetric argument implies that each column of F^* has the same sum as the corresponding column of F. Every entry $F^*[i, j]$ is either 0 or 1. Thus, F^* is a legal rounding of F.

- On the other hand, suppose F^* is a legal rounding of F. Then for each row index i, we have

$$\begin{aligned} f^*(s \to r_i) &= \sum_k f^*(r_i \to c_j) && \text{[conservation at } r_i\text{]} \\ &= \sum_k F^*[i,k] && \text{[definition of } F^*[i,k]\text{]} \\ &= \sum_k F[i,k] && \text{[definition of rounding]} \\ &= c(s \to r_i) && \text{[definition of } c(s \to r_i)\text{]} \end{aligned}$$

  We conclude that f^* saturates every edge leaving s. A symmetric argument implies that f^* saturates every edge entering t.

Our network has $O(m+n)$ vertices and $O(mn)$ edges. Thus, if we compute the maximum flow using Orlin's algorithm, our algorithm runs in $O(VE) = \boldsymbol{O(mn(m+n))}$ ***time***. Alternatively, the value of the maximum flow is at most $O(mn)$, so Ford-Fulkerson finds the maximum flow in $O(|f^*| \cdot E) = O(m^2 n^2)$ time. ■

Rubric: 7 points: standard graph-reduction rubric (scaled-ish). This is not the only correct proof. This is more detail than necessary for full credit. The running time must be reported as a function of the input parameters n and m.

(b) Prove that a legal rounding is possible *if and only if* the sum of entries in each row is an integer, and the sum of entries in each column is an integer. In other words, prove that either your algorithm from part (a) returns a legal rounding, or a legal rounding is *obviously* impossible.

Solution: One direction is trivial: If any row or column has a non-integer sum, then there is no legal rounding. So suppose every row sum and column sum is integral.

Consider the flow f defined by setting $f(r_i \to c_j) = F[i,j]$ for every row i and column j, and saturating every edge $s \to r_i$ and $c_j \to t$. Straightforward definition-chasing implies that f is indeed a flow with value $\sum_{i,j} F[i,j]$. Moreover, f is actually a *maximum* flow, because it saturates every edge leaving s.

It follows that any *integral* maximum flow f^* also has value $\sum_{i,j} F[i,j]$, because all maximum flows have the same value. Thus, f^* also saturates every edge leaving s and every edge entering t. Integrality implies that every flow value $f^*(r_i \to c_j)$ is either 0 or 1. Our argument in part (a) now implies that F^* is a legal rounding of F. ■

Rubric: 3 points = 1 for trivial case + 2 for non-trivial case. "See part (a)" is worth full credit if the proof in part (a) actually solves part (b) as well.

2. Describe and analyze an algorithm to compute the maximum number of people that can walk from the Earth fountain to the Fillory fountain in at most h hours, without anyone alerting the Beast or turning into a niffin.

Solution: Assume that the input graph G is a *directed* graph. If not, replace each undirected edge with a pair of opposing directed edges. This transformation allows two people x and y to pass through the same gate in opposite directions at the same time. However, if there is an optimal solution where that happens, there is another optimal solution where x and y do not pass through that gate, then x follows the rest of y's optimal path, and y follows the rest of x's optimal path. Thus, any optimal solution for the directed-graph problem can be transformed into an optimal solution for the undirected-graph problem.

Given the input graph $G = (V, E)$ and the integer h, we define a new directed graph $G' = (V', E')$ as follows:

- V' contains all pairs (v, i), where $v \in V$ and i is an integer between 0 and h. Intuitively, (v, i) denotes "fountain v after i hours".
- E' contains two types of edges:
  - An edge $(u, i-1) \to (v, i)$ with capacity 1, for every edge $u \to v \in E$ and every index i. Traversing this edge corresponds to passing through the gate $u \to v$ during the ith hour.
  - An edge $(v, i-1) \to (v, i)$ with capacity ∞, for every vertex $v \in V$ and every index i. Traversing this edge indicates staying in plaza v (not passing through any gate) during the ith hour.

We can build G' in $O(V' + E') = O(hV + h(V + E)) = O(hE)$ time by brute force.

We now need to find the value of the maximum flow in G' from $(\mathit{Earth}, 0)$ to $(\mathit{Fillory}, h)$. We can compute this value in $O(V'E')$ time using Orlin's algorithm. Altogether our algorithm runs in $O(V'E') = O(hV \cdot h(V + E)) = \boldsymbol{O(h^2VE)}$ ***time***.

Alternatively, because there are at most hV edges of the form $(\mathit{Earth}, i) \to (v, i+1)$, each with unit capacity, the value of the maximum flow in G' is at most hV. Thus, we can compute the maximum flow using Ford-Fulkerson in $O(|f^*| \cdot E') = \boldsymbol{O(h^2VE)}$ ***time***.

Now we need to prove that the number of people who can safely walk from Earth to Fillory is equal to the maximum flow value in G'. As usual, the proof has two parts.

[$\geq$] Let f^* be a maximum flow in G'. Because all finite capacities are integers, we can assume without loss of generality that $f^*(e)$ is an integer for every edge $e \in E'$. By the flow decomposition theorem, we can decompose f^* into $|f^*|$ paths, each carrying one unit of flow, and possibly one or more cycles, which we can discard. Each path has the form

$$(v_0, 0) \to (v_1, 1) \to (v_2, 2) \to \cdots \to (v_{h-1}, h-1) \to (v_h, h)$$

where $v_0 = \mathit{Earth}$ and $v_h = \mathit{Fillory}$, and for each index i, either $v_i = v_{i+1}$ or $v_i \to v_{i+1}$ is an edge in E.

- A person can follow this path from the Earth fountain to the Fillory fountain without turning into a niffin: For each index i, if $v_{i-1} \neq v_i$, they go through the gate from fountain v_{i-1} to fountain v_i during the ith hour; otherwise, they stay at fountain $v_{i-1} = v_i$ for the ith hour.

- Because f^* is feasible, at most one flow path goes through each gate edge $(u,i)\to(v,i+1)$, and therefore at most one person goes through the gate $u\to v$ during hour i, so the Beast is never alerted.

We conclude that ***at least*** $|f^*|$ people can safely walk from Earth to Fillory.

[$\le$] Suppose on the other hand that k people can safely walk from Earth to Fillory. We can record each person's trajectory as a path in G' of the form

$$(v_0,0)\to(v_1,1)\to(v_2,2)\to\cdots\to(v_{h-1},h-1)\to(v_h,h)$$

where $v_0 = \mathit{Earth}$ and $v_h = \mathit{Fillory}$ and for each index i, either $v_i = v_{i+1}$ or $v_i\to v_{i+1}$ is an edge in E. Specifically:

- If the person walks through the gate from fountain u to fountain v in the ith hour, the path contains the edge $(u,i-1)\to(v,i)$—which must be an edge in G' because nobody turned into a niffin.
- If the person stays next to fountain v during the ith hour, the path contains the edge $(v,i-1)\to(v,i)$.

Because the Beast is never alerted, at most one of these paths goes through any gate edge $(u,i-1)\to(v,i)$. Thus, summing these k paths yields a feasible flow in G' from $(\mathit{Earth},0)$ to $(\mathit{Fillory},h)$; the resulting flow has value k. We conclude that the maximum flow value in G' is ***at least*** k.

We conclude that the number of people who can walk from the Earth fountain to the Fillory fountain in h hours is equal to the the value of the maximum flow in G' from $(\mathit{Earth},0)$ to $(\mathit{Fillory},h)$. ■

Rubric: 10 points: standard graph-reduction rubric

Solution (minimum-cost flows): As in the previous solution, assume G is a *directed* graph. Let's define a ***canonical*** solution to be any solution that independently sends as many people as possible through a set of edge-disjoint paths $P_1, P_2, \ldots$ from *Earth* to *Fillory*; specifically, for each index i, we send $\max\{0, h+1-|P_i|\}$ people along P_i. Every instance of the Fillory problem has an optimal solution that is canonical; this observation follows from a classical result of Ford and Fulkerson.[1]

We can compute the best canonical solution as follows. Assign each edge in G a capacity of 1 and a cost of 1. Add a new edge $\mathit{Fillory}\to\mathit{Earth}$ with infinite capacity and cost $-(h+1)$, and call the resulting graph H. Compute a minimum-cost circulation ϕ in H, and return the negation of the cost of ϕ.

Let ϕ be any minimum-cost circulation in H. The flow-decomposition theorem implies that ϕ is a weighted sum of cycles. Because every edge in G has capacity 1 and positive cost, these cycles all contain the edge $\mathit{Fillory}\to\mathit{Earth}$ but are otherwise edge-disjoint, and each cycle carries exactly one unit of flow. Thus, ϕ is actually an unweighted sum of cycles: $\phi = \sum_i C_i$.

[1] No, not that one. Ford and Fulkerson's proof uses linear-programming duality, which is a significant generalization of the maxflow-mincut theorem.

Now let f be the restriction of ϕ to the original graph G; this function is a flow from Earth to Fillory. For each index i, let P_i be the directed path from Earth to Fillory in C_i. The paths P_i are edge-disjoint. Finally, for each i let ℓ_i denote the length of P_i.

The total cost of C_i is exactly $\ell_i - (h+1)$. If any cycle C_i has non-negative total cost, we can discard it without increasing the cost of f; it follows that (without loss of generality) $\ell_i \leq h$ for all i. Thus, the number of people that can walk from Earth to Fillory along P_i is exactly $(h+1) - \ell_i \geq 1$. We conclude that the total number of people that can walk from Earth to Fillory following the paths $P_1, P_2, \ldots$ is the negation of the cost of ϕ. Thus, minimizing the cost of ϕ maximizes the number of people that can walk from Earth to Fillory.

We can compute the minimum-cost circulation ϕ using Orlin's algorithm in $O(E^2 \log^2 V)$ time. Alternatively, we can use the easier cycle-canceling algorithm, always canceling the shortest cycle through *Fillory*→*Earth*. We can find this cycle by finding the shortest path in the current residual graph from *Earth* to *Fillory*. By induction, every negative cycle in the residual graph contains *Fillory*→*Earth*, so when there are no more residual paths from Earth to Fillory, there are no more negative residual cycles. We can compute each shortest path in $O(VE)$ time using Bellman-Ford, and each iteration increases the number of edges incident to *Earth* that carry flow, so the algorithm ends after at most V iterations. Altogether the algorithm runs in ***$O(V^2E)$ time***.[2] ■

[2] Using dual prices and edge reweighting, together with an implementation of Dijkstra's algorithm using Fibonacci heaps, the running time can be improved to $O(V^2 \log V + VE)$.