Name: ____________________ Wisc id: ____________________

Assembling the Night's Watch Recruits

The Night's Watch, guardians of the realm against the dangers beyond the Wall, is seeking to enroll new recruits from various regions of Westeros. Currently, there are m potential recruits from n different regions. Let m_j denote the number of recruits available from region j, for $j \in \{1, \dots, n\}$.

The Night's Watch has q castles along the wall, each with a capacity of p_i, for $i \in \{1, \dots, q\}$. Each castle will host a group of recruits. To ensure a balanced defense and promote unity among the recruits, the Night's Watch has a rule that no more than ℓ recruits from the same region can be in the same castle. The Night's Watch aims to maximize the number of recruits that can be enrolled.

Smiths of Flea Bottom's Nestling Dilemma

In the bustling city of King's Landing, a small but renowned workshop known as *Smiths of Flea Bottom* specializes in crafting exquisite armor and weapons for the noble houses of Westeros. A recent shipment of rare and precious metals has arrived at the workshop, each piece encased in its own sturdy crate. As the workshop is cramped and cluttered with all manner of blacksmithing tools and materials, space is at a premium.

The crates containing the metals vary in size, as they hold different types of metals for different purposes. The workers at the workshop spent a morning trying to figure out how to store all these crates, realizing that they could save space if some of the crates could be nested inside others.

Suppose each crate i is a rectangular parallelepiped with side lengths equal to (i_1, i_2, i_3). Geometrically, one crate can nest inside another if it can be rotated so that it fits inside the larger in each dimension. Formally, we can say that crate i with dimensions (i_1, i_2, i_3) nests inside crate j with dimensions (j_1, j_2, j_3) if there is a permutation $\{\alpha, \beta, \gamma\}$ of the dimensions $\{1, 2, 3\}$ such that $i_\alpha < j_1$, $i_\beta < j_2$, and $i_\gamma < j_3$. Nesting is recursive: if i nests in j, and j nests in k, then by putting i inside j inside k, only crate k is visible.

A crate is visible in a nesting arrangement if it is not nested inside another crate. The goal for the workers at Smiths of Flea Bottom is to choose a nesting arrangement that minimizes the number of visible crates, as only the visible crates are taking up valuable space in the workshop.

Night King's Invasion Strategy

From the icy desolation of the Lands of Always Winter, the Night King marshals his forces for an assault on the realm of Westeros. With an army of wights xat his command, he prepares to spread his dominion southward, starting with the formidable defenses of the Wall.

The Night King must devise an invasion plan to overcome the defenses of the Wall and march on King's Landing. The Wall is dotted with forts, each connected to various towns, forts, and seats of power in Westeros. The Night King realizes that sending too many wights through a particular road or town/fort/seat is risky, so he's setting a limit on the number of wights going through a road from town i to town j as R_{ij} and the limit on the number of wights going through a town/fort/seat i as L_i. Your goal is to calculate the maximum number of wights the Night King can send to King's Landing.

You may assume that The Lands of Always Winter and King's Landing have no capacity on wights and they both would be a node in the graph network.

Network flow recap

Flow network

- A directed graph $G = (V, E)$, where $|V| = n$ and $|E| = m$.
  - Each edge $e \in E$ has a *capacity* $c_e \geq 0$.
  - A *source* node $s \in V$, and a *sink* node $t \in V$.
  - All other nodes ($V \setminus \{s, t\}$) are internal nodes.
  - Let $C = \sum_{e \text{ out of } s} c_e$.
- *Flow function*: $f : E \to R^+$, where $f(e)$ is the flow across edge e.
  - Valid flow function conditions:
    i *Capacity*: For each $e \in E, 0 \leq f(e) \leq c_e$.
    ii *Conservation*: For each $v \in V \setminus \{s, t\}$, $\sum_{e \text{ into } v} f(e) = f^{\text{in}}(v) = f^{\text{out}}(v) = \sum_{\text{ out of } v} f(e)$.
- The flow starts at s and exits at t.
- Flow value $v(f) = f^{\text{out}}(s) = f^{\text{in}}(t)$.

An $s - t$ cut

- A Cut: Partition of V into sets (A, B) with $s \in A$ and $t \in B$.
- Cut capacity: $c(A, B) = \sum_{e \text{ out of } A} c_e$.

Residual Graph: Given a flow network G and a flow f on G, we define the residual graph G_f:

- Same nodes as G.
- For edge (u, v) in E:
  - Add edge (u, v) with capacity $c_e - f(e)$.
  - Add edge (v, u) with capacity $f(e)$.

Ford-Fulkerson Algorithm: $O(m\,C)$

- Initialize $f(e) = 0$ for all edges.
- While G_f contains an augmenting path P:
  - Update flow f by BOTTLENECK(P, G_f) along P.

Other algorithms:

- Scaled Ford-Fulkerson: $O(m^2 \log C)$.
- Fewest Edges Augmenting Path [BFS] (Edmonds-Karp): $O(m^2 n)$.
- Dinitz 1970: $O\left(\min\left\{n^{\frac{2}{3}}, m^{\frac{1}{2}}\right\} m\right)$.
- Preflow-Push 1974/1986: $O(n^3)$.
- Best: Orlin 2013: $O(mn)$