Assume that you are the manager of a security company tasked with assigning security officers (n) to a group of companies (m) in a certain month with 30 days numbered $\texttt{D}_0, \texttt{D}_1, \ldots, \texttt{D}_{29}$. Assume that security officers are denoted by $\texttt{SO}_0, \texttt{SO}_1, \ldots \texttt{SO}_{n-1}$ and the companies are represented by $\texttt{C}_0, \texttt{C}_1, \ldots \texttt{C}_{m-1}$. There are three 8-hour shifts per day: $\texttt{S}_0$ (midnight-8am), $\texttt{S}_1$ (8am-4pm), and $\texttt{S}_2$ (4pm-midnight). For each pair company/shift $(\texttt{C}_j, \texttt{S}_k)$, the company $\texttt{C}_j$ has a different number of security officers required for the shift $\texttt{S}_k$, however, this is constant for each day of the month.

The security officers specify the shifts they are interested in working on, and they have the opportunity to indicate more than one (1) preferred shift. However, they cannot be allocated to more than one (1) shift per day. Further, the shift preference of each security officer should be the same for all the days of the month (and therefore specified only once per security officer).

Your task is to provide a plan for assigning officers to the companies for the coming month according to the requirement of the number of security officers per shift by each company. For each day of the month, and for each shift, each company should be allocated exactly the same number of security officers it requests. There are 2 integer parameters, $0 \leq$`min_shifts`≤ 30 (minimum number of shifts) and $0 \leq$`max_shifts` ≤ 30 (maximum number of shifts), and the total number of shifts allocated to each security officer in a month should be within these two integer parameters (closed interval).

As long as you comply with the constraints above, you are free to allocate the security officers any way you want.

The input to your function `allocate` consists of a list of preferred shifts provided by each security officer and the number of security officers requested by the companies for each shift: `preferences` is a list of lists, where `preferences[i][k]`, is a binary value (0 or 1), indicating whether security officer $\texttt{SO}_i$ is interested in working on shift $\texttt{S}_k$ in that month (1 indicates that the officer is interested). `officers_per_org` is a list of lists, where `officers_per_org[j][k]` is a non-negative number that specifies how many security officers company $\texttt{C}_j$ needs for shift $\texttt{S}_k$ on each day of that month.

Your task is to implement a function `allocate(preferences, officers_per_org, min_shifts, max_shifts)` which returns:

- `None` (i.e., Python NoneType), if no allocation satisfying all constraints exists.
- Otherwise, it returns a list of lists `allocation`, where `allocation[i][j][d][k]` is equal to 1 if security officer $\texttt{SO}_i$ is allocated to work for company $\texttt{C}_j$ during shift $\texttt{S}_k$ on day $\texttt{D}_d$.

Complexity requirement: The worst-case time complexity of your solution should be $O(m * n * n)$.