1 C. Чай

В одном из отделов крупной организации работает n человек. Как практически все сотрудники этой организации, они любят пить чай в перерывах между работой. При этом они достаточно дисциплинированы и делают в день ровно один перерыв, во время которого пьют чай. Для того, чтобы этот перерыв был максимально приятным, каждый из сотрудников этого отдела обязательно пьет чай одного из своих любимых сортов. В разные дни сотрудник может пить чай разных сортов. Для удобства пронумеруем сорта чая числами от 1 до m. Недавно сотрудники отдела купили себе большой набор чайных пакетиков, который содержит a_1 пакетиков чая сорта номер 1, a_2 пакетиков чая сорта номер 2, ..., a_m пакетиков чая сорта номер m. Теперь они хотят знать, на какое максимальное число дней им может хватить купленного набора так, чтобы в каждый из дней каждому из сотрудников доставался пакетик чая одного из его любимых сортов. Каждый сотрудник отдела пьет в день ровно одну чашку чая, которую заваривает из одного пакетика. При этом пакетики чая не завариваются повторно. $1 \leq n, m \leq 50$.

2 Описание алгоритма

Составим ориентированный двудольный граф: левая доля будет состоять из сотрудников, правая из сортов чая, ребра между ними - предпочтения сотрудников. Добавим еще две вершины - source и target. source соединим со всеми вершинами из левой доли, target со всеми вершинами из правой доли. Пропускную способность ребер, направленных в target (обозначим эту пропускную способность как a_i) будет равен количеству пакетиков соответствующего чая. Пропускная способность любого ребра между долями будет равна ∞.

Опишем дальнейший алгоритм:

- Будем искать максимальный поток для чисел вида 2^k, начиная с $k = 1$, обозначив пропускную способность ребра из source в вершину сотрудника равную 2^k

- Если этот поток равен $n * 2^k$, то выполняем итерацию для $k + 1$.

- Если он меньше, то запускаем бинарный поиск на полуинтервале

$[2^{k-1}, 2^k)$ с условием $\leq answer \equiv f_l = n * l$, где l - число в полуинтервале, f_l - поток, соответствующий l

- Находим наибольшее такое l, что $f = n * l$. Это искомый ответ.

Максимальный поток ищем с помощью алгоритма Эдмондса-Карпа. Описание и корректность можно посмотреть тут .

3 Доказательство корректности работы

Предположим, что максимальное количество дней равно l, но условие $f_l = n * l$ не выполняется. Значит существует такой n_i, что $f(0, n_i) < l$. Тогда в остаточной сети существует увеличивающий поток через n_i, а значит это не максимальный поток - противоречие. Если $f_z = n * z$ для z > l, то построено распределение пакетиков такое, что его хватит на z дней, что тоже неверно - противоречие.

Корректность алгоритма поиска потока см. в пункте 2.

4 Время работы и доп. память

- V = n + m + 2, E = n + m + e, где e - количество предпочтений у сотрудников, K - искомое количество дней, $K \leq [\frac{\sum_{i=1}^{n} a_i}{n}]$
- Время работы $O(V * E^2 * logK)$
- Доп. память $O(V^2)$

5 Доказательство времени работы

См. ссылку из пункта 2. Доп. память расходуется на поддержание остаточной сети и запись пропускных способностей(с помощью таблицы смежности).