

Задача А. Торговый центр

Автор задачи и разработчик: Николай Будин

Если Кендис воспользуется i -м автобусом, она приедет в ТЦ спустя $t_i + l_i$ минут. Ответом является минимум из этих значений.

Задача В. Переливание жижи

Автор задачи и разработчик: Даниил Орешников

Эту задачу можно было решить обычным поиском в ширину. Поначалу кажется, что возможных состояний порядка $n_1 \cdot n_2 \cdot n_3$, но эту оценку можно сильно уменьшить, заметив, что

1. уже после первого переливания всегда найдется либо пустой, либо полный бак
2. сумма объемов жижи в баках не меняется

Заметим, что в таком случае состояние баков можно описать четырьмя параметрами: номер пустого или полного бака (3 варианта), пустой этот бак или полный (2 варианта), номер еще одного бака (2 варианта) и количество жижи в нем (не более $\max(n_1, n_2, n_3)$ вариантов). Итого — потенциальных состояний системы из трех баков не более $12 \cdot \max(n_1, n_2, n_3)$, так что работающий за линейное время поиск в ширину позволит найти минимальное число переливаний для достижения требуемого состояния.

Важно обработать случай, когда b_1 , b_2 и b_3 соответствуют исходным объемам и не достижимы после первого же переливания, а также не забыть о том, что порядок b_i не важен, и соответствующих им конечных состояний несколько.

Задача С. Игра в Мафию

Автор задачи и разработчик: Григорий Хлытин

Заметим, что мафией могут быть только выжившие после всех ночей игроки, и по условию их количество равно $k - m$ ($1 \leq k - m \leq 15$). Далее будем называть их подозреваемыми. Давайте переберем всевозможные подмножества подозреваемых, для каждого такого подмножества проверим, могли ли игроки из этого подмножества убить всех жертв. После этого выберем минимальное по размеру подмножество из всех прошедших проверку — это и будет ответ на задачу.

Заметим, что для такой проверки нам нужна лишь информация, с кем из подозреваемых контактировала каждая жертва в ночь убийства. В качестве одного из возможных решений можно было построить таблицу «подозреваемые-жертвы» размера $(k - m) \cdot m$, а затем проверить, что каждая жертва пересекалась хотя бы с одним из выбранных подозреваемых.

Такое решение работает за время $O(2^{k-m} \cdot (k - m) \cdot m)$.

Задача D. Японский кроссворд

Автор задачи и разработчик: Николай Будин

В задаче дано поле размера $n \times m$ и требуется в каждой строке и каждом столбце найти длины отрезков из «#». Чтобы найти длины отрезков в строке, нужно пробежаться по ее символам слева направо. Каждый раз, когда встречается символ «#», если предыдущий символ тоже равнялся «#», то нужно увеличить длину последнего отрезка на 1, а иначе нужно добавить новый отрезок длины 1.

Задача E. Минимальная строка

Автор задачи: Даниил Орешников, разработчик: Ильдар Загретдинов

Ответом является лексикографически минимальная строка длины $|a|$, составленная из символов из строк a и b . Действительно, давайте на первое место поставим минимальный из символов, представленных в строках. Если он находится в строке b , то поменяем первый элемент из a с этим элементом. Если минимальный элемент находится в строке a , то поменяем его с любым элементов из b и аналогично предыдущему варианту, поменяем его с первым символом из a . Повторяя такое действие для всех позиций первой строки, мы получим лексикографически минимальную строку.

Задача F. Ксероксинатор

Автор задачи: Григорий Шовкопляс, разработчик: Ильдар Загретдинов

Давайте хранить очередь, где каждый элемент — это пара из количества элементов, добавленных в секунду i , и самой секунды i . Тогда, на каждой итерации мы добавляем в очередь пару, а затем вынимаем из начала очереди b элементов, пересчитывая ответ. Ответ можно пересчитать, зная количество людей в стоящей в начале очереди группе и время их добавления в очередь.

После того, как итерации окончились, пересчитаем ответ с учетом оставшихся элементов (можно считать, что мы вытаскиваем все оставшиеся элементы на секунде $n + 1$).