

Задача А. Соки

Апельсиновый сок расходуется последним, поэтому его останется $\min(x, c)$ (не больше, чем его было изначально, и чем осталось). Тогда яблочного и вишнёвого сока суммарно останется $y = x - \min(x, c)$. Аналогично вишнёвого сока останется $\min(y, b)$. Тогда яблочного сока осталось $x - \min(x, c) - \min(y, b)$.

Задача В. Хоровод

Для начала разберём случай $x = y$. Понятно, что максимальное количество пар соседей разного пола достигается при чередовании мальчиков и девочек: BGBGBG...BG. В данном случае количество пар равно $x + y$. Минимальное количество пар достигается, если все мальчики идут подряд, а потом все девочки идут подряд: BB...BBGG...GG, тогда количество пар равно 2.

Заметим, что количество пар соседей разного пола всегда чётно. Чтобы доказать это, можно рассмотреть блок подряд идущих мальчиков, с ним образуется ровно 2 пары с девочками (слева и справа). Если просуммировать это по всем блокам мальчиков, то мы получим ровно количество пар соседей разного пола, и оно будет чётным.

Теперь, чтобы получить меньше, чем $x + y$ пар соседей разного пола, можно начать объединять детей одного пола в блоки. Например:

- BGBGBGB — 8 пар;
- GGBBGBGB — 6 пар;
- GGGBBBGB — 4 пары;
- GGGGBBBB — 2 пары.

То есть, мы каждый раз берём следующего мальчика и девочку и добавляем в блоки подряд идущих мальчиков и девочек, уменьшая количество пар на 2. Так можно получить любое чётное количество пар от 2 до $x + y$.

Теперь перейдём к случаю $x \neq y$. Понятно, что у нас не могут просто чередоваться мальчики и девочки. Однако максимальное количество пар достигается, когда они почти чередуются: BGBGBG...BGBGGGGG (в случае, если девочек больше). Количество пар соседей разного пола в этом случае равно $2 \cdot \min(x, y)$. Чтобы получить какое-то промежуточное количество пар (между 2 и $2 \cdot \min(x, y)$), можно так же, как описано выше, объединять мальчиков и девочек в блоки.

Задача С. Лавка с числами

Сначала разберём отдельный случай: с помощью суммы чисел на отрезке можно представить любое число, если $l = 1$. В противном случае в виде суммы нельзя представить число 1. То есть ответ — -1 тогда и только тогда, когда $l = 1$.

Далее, число k представимо в виде суммы x чисел из отрезка, если выполняется $l \cdot x \leq k \leq r \cdot x$. А значит число не представимо, если для какого-то x имеем $r \cdot x < k < l \cdot (x + 1)$, то есть x слагаемых — мало, а $x + 1$ — много. Значит, нам нужно найти максимальный «пропуск» такого вида.

Имеем неравенство, где x — количество слагаемых: $r \cdot x + 1 < l \cdot (x + 1)$ (так как между ними должно быть хотя бы одно число). Решая его, получаем $x < \frac{a-1}{b-a}$. А так как нам нужен максимальный целый x , то он будет равен $\lfloor \frac{a-2}{b-a} \rfloor$. Мы нашли максимальный «пропуск», а максимальное число из него равняется $l \cdot (\lfloor \frac{a-2}{b-a} \rfloor + 1) - 1$.

Задача D. Крайний Банк

Пусть мы знаем все пары покупка-продажа, как трейдер дошел до состояния, описанного во входных данных. Заметим, что если есть пара вида (i, j) и в момент j у нас была какая-то незакрытая сделка, в которой стоимость покупки больше чем a_i , но при этом меньше a_j , то мы можем поменять моменты покупки в этих парах и всё останется корректным.

Из этого замечания сразу же следует жадный алгоритм для решения задачи. Давайте идти по моментам времени и если мы находимся в моменте где трейдер продавал, то будем ставить ему в

пару максимальный незанятый момент покупки, стоимость товара в котором не превышает текущей стоимости продажи. Таким образом мы либо получим корректное разбиение на пары, либо в какой-то момент столкнемся с противоречием (не будет подходящего момента покупки), что значит, что трейдер никак не мог исполнить условие, для того, чтобы быть потенциальным мошенником.

Для эффективной реализации данного решения предлагается для каждого значения a_i массив с ещё не занятыми моментами покупки, имеющими такую стоимость. Если нам встретится момент продажи, то нам нужно будет найти максимальный, по стоимости, не занятый момент покупки, не превышающий данного a_i . Это легко можно сделать используя массивы, который мы поддерживаем (достаточно пройти от $a_i - 1$ до 1 и найти первый непустой массив). После этого мы создаем пару с любым элементом из этого массива и данным индексом. Удаляем данный индекс из этого массива и продолжаем процесс. Заметим, что для эффективной реализации следует всегда брать в пару последний элемент из массива, так как его удаление будет работать за $O(1)$. Таким образом мы получаем решение за $O(n)$.

Задача Е. Максимальная попарная разница

Если раскрыть все модули в $|a_1 - b_{p_1}| + |a_2 - b_{p_2}| + \dots + |a_n - b_{p_n}|$, то n чисел из $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$ войдут в сумму со знаком плюс, а оставшиеся n чисел со знаком минус.

Поэтому, максимальный счёт точно не может быть больше чем «сумма n максимальных чисел среди $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$ минус сумма n минимальных чисел среди $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$ ». И оказывается, что такого счёта всегда можно достичь.

Давайте отсортируем все числа $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$ и посмотрим: сколько чисел из массива a окажется в первой половине отсортированного массива, пусть x . Тогда чисел из массива b в первой половине всего $n - x$ штук. А во второй половине: $n - x$ чисел из массива a , и x чисел из массива b . Поэтому мы можем образовать n пар: «число из массива a , число из массива b » так, что в каждой паре по одному числу из каждой половины отсортированного массива. И счёт в таком случае, будет в точности равен «сумма n максимальных чисел среди $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$ минус сумма n минимальных чисел среди $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$ ».

А число способов это сделать будет $x! \cdot (n - x)!$, где $k! = 1 \cdot 2 \cdot \dots \cdot k$. Так как нужно независимо сопоставить x наименьшим числам из $a - x$ наибольших чисел из b , и независимо $n - x$ наибольшим числам из $a - n - x$ наименьших чисел из b .

При этом при любом другом способе, хотя бы одна из пар (a_i, b_{p_i}) будет содержать пару чисел из одной и той же половины отсортированного массива, и, так как все числа различны, в таком случае счёт будет не максимален.

Асимптотика решения: $O(n \log n)$, из-за сортировки массива.