Разбор задачи «Таблица результатов»

Заметим, что для каждой задачи можно отдельно проверить решена она или нет. Для этого в строке должен быть хотя бы один символ «+».

Тогда посчитаем число таких строк и это и будет ответ.

Разбор задачи «Ральф и арифметика»

Пусть l_n — длина десятичной записи числа n.

Посчитаем количество чисел без второстепенных цифр длины $l = 1, 2, \dots l_n - 1$. Всего у нас k второстепенных цифр, значит всего у нас $10 - k$ разрешенных цифр. Также первая цифра не может быть нулем, значит вариантов первой цифры либо $f = 10 - k$, если 0 — второстепенная цифра, либо $f = 9 - k$ иначе. Тогда ответ для фиксированной длины l равен $f \cdot (10 - k)^{l-1}$.

Теперь посчитаем количество чисел без второстепенных цифр длины l_n. Все такие числа не превосходят n. Рассмотрим число n отдельно: прибавим 1 к ответу, если в записи числа n нет второстепенных цифр. Осталось рассмотреть числа длины l_n, строго меньшие n. У таких чисел есть несколько общих старших цифр с числом n (возможно, 0), затем идет цифра, меньшая соответствующей цифры числа n, а затем любые другие цифры. Переберем количество общих старших цифр с числом n: $p = 0 \dots l_n - 1$. Если среди первых p цифр числа n есть второстепенная, то таких чисел нет. Иначе нужно найти s — количество цифр, не являющихся второстепенными, строго меньшие p-й цифры числа n, и прибавить к ответу $s \cdot (10 - k)^{l_n - p - 1}$.

Разбор задачи «Битовый автомат»

Пусть a_{min} — это минимально возможный реальный урон, а a_{max} —максимально возможный реальный урон.

Заметим, что если $a = 0$, то оно минимально возможное, а если число $a = 2^n - 1$, то оно максимально возможное. Соответственно, для первого случая $a_{min} = a$, а для второго — $a_{max} = a$.

Если a не минимальное, то чтобы найти такое a_{min}, которое получается из a изменением одного бита, необходимо заменить самую старшую единичку в битовом представлении на ноль, поскольку это гарантирует максимальную разницу между a и a_{min}. Аналогично, чтобы получить максимальную разницу между a_{max} и a, необходимо заменить самый старший ноль битового представления a на единицу. При этом необходимо учитывать, что битовое представление числа a фиксированно и имеет длину n.

Пройдём по всем n битам числа a и посчитаем таким образом a_{min} и a_{max}.

Разбор задачи «Монетки»

Применим алгоритм, похожий на решето Эратосфена. Возьмем последовательность $a_i = i^2 + 1$: 2, 5, 10, 17, 26, 37, 50, 65, …

Возьмем первое число — $2 = 1^2 + 1$. Посмотрим, какие числа на него делятся (для них все 2 – это ответ) — это числа $(1 + 2 \cdot 0)^2 + 1$, $(1 + 2 \cdot 1)^2 + 1$, $(1 + 2 \cdot 2)^2 + 1$, …. Поделим все эти числа на 2 (на максимальную степень двойки, на которую они делятся), получим обновленную последовательность: 1, 5, 5, 17, 13, 37, 25, 65, …

Теперь возьмем следующее число – $5 = 2^2 + 1$. Посмотрим, какие числа делятся на него — это $(1 + 5 \cdot 0)^2 + 1$, $(1 + 5 \cdot 1)^2 + 1$, $(1 + 5 \cdot 2)^2 + 1$, … Поделим их все на 5 (тоже на максимальную степень 5, на которую они делятся): 1, 1, 5, 17, 13, 37, 1, 65, …

Теперь возьмем третье число — 5, опять проделаем ту же самую операцию, получим: 1, 1, 1, 17, 13, 37, 1, 13, …

Несложно доказать, что каждый очередной делитель (1, 5, 5, …), который мы берем — простое число или 1. Таким образом, мы находим все простые делители чисел вида $n^2 + 1$.

Алгоритм работает за $O(n + n/2 + n/3 + ... + n/n)$, что равно $O(n \log(n))$

Разбор задачи «Доставка»

Легко понять, что стоимость товара с учетом доставки из магазина с номером i составит $p_i + d_i + C \cdot v_i$ очков. Значит, чтобы получить ответ на задачу, надо выбрать минимальное значение этой величины по всем i.

Разбор задачи «Уничтожение дронов»

Для каждого дрона оценим через сколько он сможет добраться до Ральфа, если будет двигаться по оптимальному маршруту. За каждый ход дрон может менять обе свои координаты на 1. Таким образом дрону с в точке с координатами x_i и y_i понадобится $|x_i|$ ходов, чтобы оказаться на прямой $x = 0$, и $|y_i|$ ходов, чтобы оказаться на прямой $y = 0$. Соответственно в точке (0, 0), в худшем для Ральфа случае, он сможет оказаться через $\max(|x_i|, |y_i|)$ ходов. Примем эту величину за расстояние дронов до Ральфа.

Заметим, что выгоднее всего для Ральфа стрелять по дронам, расстояние до которых меньше. Отсортируем дронов по увеличению расстояния.

В каком случае Ральф не успеет уничтожить i-го дрона? В том случае, если расстояние до него будет меньше, чем число дронов с расстоянием не больше чем у него. Таким образом пройдемся по дронам в порядке увеличения расстояния и проверим, что $\max(|x_i|, |y_i|) >= i$ (нумерация дронов с 1). Если для всех дронов это выполнено, то можно вывести их в порядке увеличения расстояний, иначе решений не существует.

Разбор задачи «Конфета в лабиринте»

Заметим, что если можно пронести леденец длины k, то можно и всех меньших длин. Поэтому сделаем бинарный поиск по ответу. Что бы проверить, можно ли пронести леденец длины k, сделаем обход в глубину, где вершинами будут пары из самой верхней левой клетки леденца и направления, которому параллелен леденец. Из каждого такого состояния есть 6 переходов — два, в направлении сдвига на одну клетку вдоль леденца, и еще повороты на 90° вокруг одного из концов. Что бы проверить, что мы не заходим на стену, можно предподсчитать префиксные суммы по каждой строке и столбцу. Итого решение за $O(n \cdot m \cdot \log(\min(n, m)))$.

Разбор задачи «Пароли»

Заметим, что условие задачи можно переформулировать следующим образом. Посчитайте количество способов разбить s на три строки a, b и c, так что $a \neq b$, $b \neq c$ и $a + b \neq b + c$.

Количество способов разбить s на три отрезка — $C^2_{|s|-1}$.

Вычтем количество способов, в которых $a = b$ или $b = c$. Чтобы посчитать количество разбиений, в которых $a = b$, переберем четную длину $l \cdot 2$ строки $a + b$, и проверим, что $s[1 \ldots l] = s[l + 1 \ldots l \cdot 2]$ (для этого можно, например, использовать z-функцию). Количество разбиений, в которых $b = c$, считается аналогично. Обратите внимание, что возможен случай, когда $a = b = c$. Его нужно учесть, и в таком случае, прибать 1 к ответу.

Осталось вычесть количество разбиений, в которых $a \neq b$, $b \neq c$ и $a + b = b + c$. Для этого, переберем длину l строки $a + b$. Проверим, что $s[1 \ldots l] = s[|s| - l + 1 \ldots |s|]$, $s[1 \ldots |s| - l] \neq s[|s| - l + 1 \ldots l]$ и $s[|s| - l + 1 \ldots l] \neq s[l + 1 \ldots |s|]$. Для этого тоже можно использовать z-функцию.