IQ-тест

В этой задаче нужно было найти в квадрате 4×4 такой квадрат 2×2, что количество черных клеток в нем не равно 2. Если это количество равно 1 или 3, то мы можем поменять один символ так, что квадрат станет полностью состоять из клеток одного цвета. Если же это количество равно 0 или 4, то квадрат уже полностью одноцветный.

Водопровод

В этой задаче надо было представить данное число $n-1$ в виде суммы чисел из набора $1, 2, \ldots, k-1$. Заметим, что любое число $x \in [1, \frac{k \cdot (k-1)}{2}]$ достижимо. Как это доказать? Будем, постепенно набирать числа с конца (если текущее набранное число меньше данного, возьмем максимальное число, такое, что сумма не превысит данное). Тогда, заметим, что если мы не можем взять максимальное неиспользованное число, то найдется число от 1 до максимального неиспользуемого, дающего в сумме с набранным ровно данное. Тогда ответа нет только тогда, когда он больше суммы всех чисел. Теперь докажем, что приеденный алгоритм использует минимально возможное количество чисел. Наш ответ представляет собой суффикс (множество $\{x, x+1, \ldots, k-1\}$ для некоторого x) всего набора чисел и, возможно, еще одно число. Тогда любой набор, в котором будет строго меньше чисел, чем в нашем, будет по сумме строго меньше, чем наш (так как наибольший по сумме такой набор — суффикс всего множества, но в найденном наборе есть, кроме суффикса, еще одно число). Тогда остается бинпоиском найти самый длинный суффикс, сумма чисел в котором меньше, либо равна n. Если сумма чисел в такой суффиксе равна n, то ответ — длина суффикса, иначе — длина суффикса, увеличенная на 1. Обратите внимание, что число 0 представимо в виде суммы пустого множества чисел.

Счастливая перестановка

Счастливой перестановкой при $n = 4$ является перестановка $[2, 4, 1, 3]$, для $n = 1$ — $[1]$. Научимся получать из перестановки для n перестановку для $(n+4)$.

- Увеличим все числа в исходной перестановке на 2
- Припишем слева 1, $n+4$, справа — $n+3$, 2.

Для чисел, которые были в исходной перестановке соотношение не изменится: $p_{p_i} = n + 1 - i$ верно $p'_{p'_i} = p_{p_{i-2}+2-2} + 2 = p_{p_{i-2}} + 2 = n + 1 - i + 2 + 2$ $p'_{p'_i} = (n+4) + 1 - i$
Для первых двух и последних двух чисел соотношение тоже выполняется.

Докажем, что решения, для $n = 4k + 2$ и $n = 4k + 3$ не существует.

- Рассмотрим множество u пар (i, p_i).
- Пусть $(a, p_a) \in u$ тогда по определению $(p_a, n + 1 - a) \in u$, тогда $(n + 1 - a, n - p_a + 1) \in u$, следовательно $(n - p_a + 1, n + 1 - (n + 1 - a)) = (n - p_a + 1, a) \in u$, отсюда $(a, p_a) \in u$.
- Тогда $\forall a$ заданы значения p_{p_a}, p_{n+1-a}, p_{n-p_a+1}. То есть, все элементы разбиваются на четверки (если числа a, p_a, $n + 1 - a$ и $n - p_a + 1$ попарно различны).
- Однако, элементы не всегда попарно различны. Пусть $a = n + 1 - a$, тогда $2a = n + 1$, то есть a — центральный элемент перестановки. Тогда нетрудно убедиться в остальных равенствах.
- Утверждается, что числа могут быть не различны только тогда, когда a — центральный элемент, то есть, все остальные элементы разбиваются на четверки.

- Рассмотрим пары (a, p_a), $(p_a, n+1-a)$, $(n+1-a, n+1-p_a)$, $(n+1-p_a, a)$. Если равны первые члены пар, то равны и вторые (по построению, для каждой пары (i, j) $p_i = j$). Можно обозначить $x = a, y = p_a, z = n+1-a, t = n+1-p_a$) и записать пары как (x, y), (y, z), (z, t), (t, x). Заметим, что из равенства двух элементов внутри пары следует равенство элементов в следующей паре (для последней пары следующая пара — первая). Тогда, если два элемента внутри любой пары равны, то элементы внутри каждой из пар равны.
- Тогда остается рассмотреть два равенства вне пар $a = n+1+a$ и $p_a = n+1-p_a$. Первое мы уже рассмотрели выше, рассмотрим второе.
- $p_a = n+1-p_a$, $p_{p_a} = p_{n+1-p_a}$, тогда $n+1-a = a$, а это равенство уже рассматривалось.
- Тогда все элементы, кроме, возможно, центрального, разбиваются на четверки.

Таким образом, есть решения для $n = 4k$ и $n = 4k+1$, их и строит показанный конструктивный алгоритм.

Сдвиги

Требовалось промоделировать следующий процесс:

- Есть перестановка $1, 2, 3, \ldots, n$
- Происходит циклический сдвиг каждого из отрезков $[1;2], [3;4], \ldots [2 \cdot k + 1; n]$ влево на 1 (k — максимальное целое число, такое, что $2 \cdot k + 1 \leq n$.
- Происходит циклический сдвиг каждого из отрезков $[1;3], [4;6], \ldots [3 \cdot k + 1; n]$ влево на 1
- Процедура повторяется для отрезков длины 4, 5, ..., n.

Посчитаем количество сдвигов отрезков в сумме за всю процедуру.

$$R = \sum_{i=1}^{i \leq n} \lceil \frac{n}{i} \rceil \leq C_1 \cdot n \cdot \sum_{i=1}^{i \leq n} \frac{1}{i} \leq C_2 \cdot n \cdot (\int_a^b \frac{dx}{x}) \leq C_3 \cdot n \cdot \log n$$

Здесь C_i — некоторые константы. Итак, мы получили, что в сумме будет не очень много сдвигов, которые нужно смоделировать. Научимся моделировать каждый сдвиг за $O(1)$.
Так будет преобразовываться перестановка из пяти элементов:

```
1 2 3 4 5
2 1 4 3 5
1 4 2 5 3
4 2 5 1 3
2 5 1 3 4
```

Обратим внимание, как происходит каждый сдвиг. Один элемент в каждом отрезке «прыгает» через все остальные. Рассмотрим сдвиг всех отрезков длины 3.

```
1 2 3 4 5 6 7 8 9
2 3 1 5 6 4 8 9 7
```

Уберем из перестановки «прыгающие» элементы.

```
* 2 3 * 5 6 * 8 9
2 3 * 5 6 * 8 9 *
```

Здесь видно, что все прыгающие элементы переходят из свое й ячейки в ячейку следующего прыгающего элемента. Однако при этом добавляется новая ячейка в конце перестановки, в которую перемещается последний прыгающий элемент.

```
* 2 3 * 5 6 * 8 9
  2 3 * 5 6 * 8 9 *
```

Тогда мы можем легко промоделировать этот процесс за время, линейно зависящее от количества прыгающих элементов. Мы получим нужную перестановку, которая будет сдвинута на 1 вправо в памяти. Постепенно этот сдвиг будет накапливаться, но не превысит n. Тогда достаточно зарезервировать для перестановки массив длины $2 \cdot n$ и $n - 1$ раз сдвинуть нужные элементы.

Утерянная последовательность

В задаче предлагались по типу скобок на каждой позиции и по позициям некоторых закрывающихся скобок восстановить какую-нибудь подходящую правильную скобочную последовательность. Заметим, что развернув данную нам последовательность $\{p_1, p_2, \ldots, p_n\}$, а также заменив все числа q_i на $(n - q_i + 1)$, мы узнаем типы скобок и позиции некоторых открывающихся скобок в некоторой правильной скобочной последовательности $\{y_1, y_2, \ldots, y_n\}$, развернув которую мы получим ответ на задачу.
Дальше будем считать, что последовательность $\{p_1, p_2, \ldots, p_n\}$ развернута, а все числа q_i заменены на $(n - q_i + 1)$.
Найти нужную последовательность $\{y_1, y_2, \ldots, y_n\}$ можно жадным алгоритмом. Будем идти слева направо по последовательности $\{p_1, p_2, \ldots, p_n\}$, поддерживая стек текущих открытых скобок. Тогда пусть мы на i-ом шаге алгоритма, а верхний элемент стека (если он не пуст) равен s. Тогда:

- Если стек пуст, или он не пуст, но $s \neq p_i$, то положим в него p_i, а также присвоим $y_i = p_i$.
- Если же стек не пуст, а $s = p_i$, то:
  - Если на данной позиции должна стоять открывающаяся скобка ($i = q_j$ для некоторого j), то положим в стек p_i, присвоим $y_i = p_i$.
  - Иначе вынем из стека s, присвоим $y_i = -p_i$.

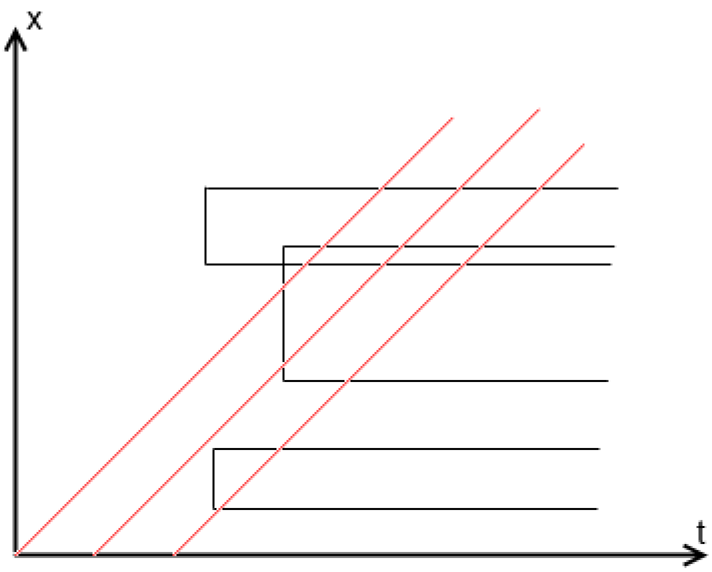
Если в конце стек оказался не пуст, то ответ на задачу "NO". Иначе $x_i = -y_{n-i+1}$ $(1 \leq i \leq n)$.
Понятно, что если мы нашли ответ, то он корректный. Докажем теперь, что если ответ существует, то наш алгоритм найдет его.
Предположим, нужная последовательность $\{x_1, x_2, \ldots, x_n\}$ существует, а наш алгоритм не нашел ответ. Тогда существует и корректная последовательность $\{y_1, y_2, \ldots, y_n\}$, а наш алгоритм получил некорректную последовательность $\{our_1, our_2, \ldots, our_n\}$. Пусть i — первая позиция, в которой они отличаются ($our_i \neq y_i$, $our_j = y_j$ для любого $1 \leq j < i$). Понятно, что $|our_i| = |y_i|$. Тогда возможно два случая:

- $our_i > 0$, а $y_i < 0$. Такого быть не может, так как если возможно, то наш алгоритм всегда поставит закрывающуюся скобку.
- $our_i < 0$, а $y_i > 0$. Тогда наш алгоритм поставил закрывающююся скобку, а в корректном ответе на данной позиции стоит открывающяяся. Тогда найдем в последовательности $\{y_1, y_2, \ldots, y_n\}$ парную скобку к i-ой. Пусть она на позиции j $(j > i)$. Тогда поменяем местами скобки на позициях i и j в последовательности $\{y_1, y_2, \ldots, y_n\}$. Заметим, что последовательность останется правильной. Она также останется корректной с точки зрения задачи, так как $|y_i| = |y_j|$, а на позиции i может стоять закрывающаяся скобка (иначе наш алгоритм ее бы не поставил).

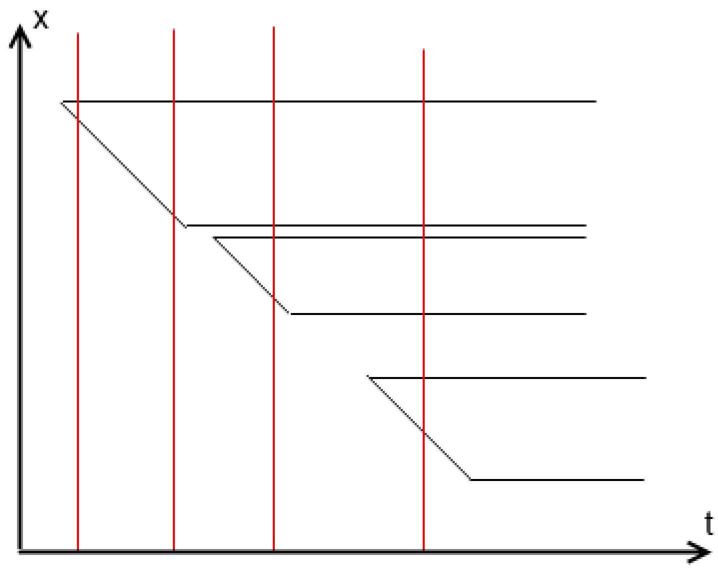
Таким образом, существует корректный ответ, который совпадает с нашей последовательностью как минимум в первых i позициях. Повторяя такую операцию, получим, что существует корректный ответ, полностью совпадающий с нашим. Тогда наш ответ корректный и предположение неверно. Значит, наш алгоритм всегда найдет ответ, если он существует.
Асимптотика решения $O(n)$.

Туристы

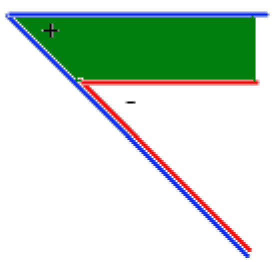
В этой задаче удобно было ввести систему координат с осью времени и осью отрезков. Тогда каждая пара будет идти по прямой $x = t - qt_i$, а каждый отрезок станет бесконечным вправо прямоугольником. Чтобы понять, сколько пара путников будет идти вдоль отрезков, нужно посчитать длину пересечения соответствующей прямой и прямоугольников.



Решим задачу, где отрезки не могут пересекаться. Тогда удобно сменить метрику следующим образом: $(x, y) \to (x - y, y)$. Тогда система отрезков и прямых будет иметь следующий вид:



Тогда нужно находить сумму длин пересечения всех скошенных прямоугольников и вертикальной прямой. Это довольно просто. Представим каждый прямоугольник, как разность двух углов величины $\frac{\Pi}{4}$.



Осталось научиться пересекать угол и прямую. Пусть вершина угла (t_0, x_0) и прямая задается уравнением $t = t_1$. Тогда длина их пересечения равна $|t_1 - t_0|$. Тогда, если есть множество из k углов с $\sum_{i=0}^{i<k} t_i = A$, то сумма длин пересечения углов с прямой равна $k * t_1 - A$, если прямая задается уравнением $t = t_1$. Чтобы получить из этого длину пересечения прямоугольников и прямой, нужно из суммы длин пересечения положительных углов вычесть сумму длин пересечения отрицательных углов. Тогда отсортируем все углы и прямые по возрастанию t и будем поддерживать величины $\sum_{i=0}^{i<k} t_i$ для углов обоих знаков, а так же количество углов обоих знаков. Тогда для каждой прямой ответ просто получается из этих величин.
Но это решение работает только если отрезки не пересекаются. Но множеству пересекающихся прямоугольников (бесконечных вправо) можно сопоставить множество непересекающихся прямоугольников, которые занимают ту же часть плоскости. При этом, размер такого множества не будет превышать $2 * m$, где m — количество прямоугольников в исходном множестве.
Построить такое множество можно, например, с помощью следующего алгоритма:

- Отсортируем все отрезки по возрастанию t_i
- Будем поддерживать set отрезков, которые открыты, причем таким образом, чтобы пересекающиеся отрезки сливались в один
- При добавлении очередного отрезка рассмотрим все отрезки из set'а, которые его пересекают (если он вложен в какой-то из отрезков, то не будем его обрабатывать). Объединим его со всеми

пересекающими его отрезками и добавим в новое множество его части, которые не покрыты отрезками set'а. Это делается простым проходом по множеству пересекающих его отрезков. Далее добавим увеличенный отрезок в set и удалим оттуда все пересекающие его отрезки.

- Каждый элемент удалится из set'а и добавится в set ровно один раз. Отсюда получаем ограничение на размер множества и время работы $O(n \log n)$.

Дамский магазин

Это была обратная задача о рюкзаке. Даны были возможные суммы, надо было восстановить предметы.

- Если некоторое достижимое число k не представляется в виде суммы некоторого набора меньших достижимых чисел, то оно обязано входить в набор предметов. Это довольно очевидное утверждение, если такого предмета нет и k не представляется в виде суммы других предметов (веса предметов тоже, очевидно, достижимы), то его нельзя набрать. Однако оно достижимо.

- Кроме того, верно, что если число k представляется в виде суммы меньших достижимых чисел и ответ существует, то существует и ответ в котором нет предмета k. Это тоже довольно легко показать. Рассмотрим набор предметов, в котором есть k. Число k представляется в виде суммы некоторых меньших достижимых чисел. Тогда, если предмет k входил в представление какого-либо достижимого числа, заменим его набором предметов, в виде которых он был представлен. Теперь предмет k фактически не нужен. Давайте удалим его. Тогда есть набор, в котором нет предмета, весом k.

- Тогда понятно решение за $O(nm)$ (n — количество достижимых сумм, m — размер рассматриваемого отрезка). Будем поддерживать булевский массив, показывающий, достижима ли данная сумма из текущего набора предметов. Если дано, что число достижимо, но оно не достижимо из предыдущих, добавим его в набор и обновим массив.

- Поймем почему это решение находит минимальный набор. Вспомним, что набор предметов может содержать только достижимые числа.

- Рассмотрим некоторый оптимальный набор opt и полученный данным решением набор our. Пусть x — минимальное число, такое, что предмет x входит в набор our, но не входит в набор opt. Если число входит в набор our, то оно не представимо в виде суммы других чисел, а значит и предметов набора opt. Тогда набор opt некорректен. Тогда не существует чисел входящих в our и не входящих в opt. Тогда $|our| \le |opt|$. Но набор opt минимален, следовательно $|our| = |opt|$. Тогда не существует и чисел, входящих в opt и не входящих в our. Получаем, что наборы совпадают, то есть существует единственный оптимальный набор, который и находит наше решение. Поймем, что если число представимо в виде суммы набора достижимых чисел, то оно представимо и в виде суммы двух достижимых чисел. Рассмотрим некоторое представление α числа k. Удалим из α какой-нибудь предмет q. Сумма набора α' достижима и равна некоторому p. Тогда $k = p + q$.

- Из этого можно получить следующее: $d_i = \sum_{j=1}^{j<i} p_j \cdot p_{i-j}$. p_i равно 1, если сумма i достижима и 0 иначе. $d_i = 0$ в случае, если i не представимо в виде суммы меньших достижимых чисел и не равно нулю иначе. Это следует из утверждения, доказанного в предыдущем абзаце (если число i пред Тогда в набор предметов войдут все такие i, что $d_i = 0$ и $p_i = 1$. Кроме того, если существует $d_i = 1$ и $p_i = 0$, ответа не существует. Заметим, что ответ будет совпадать с ответом решения за $O(nm)$.

- Однако d_i не обязательно считать за $O(m^2)$. Можно заметить, что как многочлен d — квадрат p как многочлена. То есть d можно найти за $O(m \log m)$ при помощи быстрого преобразования Фурье.