Задача А. Неклассические классики

Заметим, что нам всегда нужно ровно $2{\cdot}k{+}2$ горизонтальных отрезков (это в точности количество строк $+1$).

С вертикальными отрезками все немного сложнее. Мы можем использовать $k \cdot (a+1) + k \cdot (b+1) + (n+1)$ отрезков, просто обведя боковые стороны каждого квадрата. К сожалению, это число не всегда минимально. Почему? Потому что иногда некоторые боковые стороны квадратов одного ряда как бы «продолжают» боковые стороны квадратов соседнего ряда. В таком случае можно два отрезка заменить на один отрезок вдвое большей длины.

Заметим, что такое не происходит, когда два соседних ряда имеют разную четность. А когда они имеют одинаковую четность, все вертикальные отрезки меньшего ряда соединены с отрезками большего ряда.

Тогда будем рисовать наше поле от первого ряда к последнему. Если мы только что нарисовали i-й ряд, в котором было a клеток, а теперь хотим нарисовать следующий ряд, в котором b клеток, то бывает три случая:

1. Четности a и b разные. Тогда нам придется поставить b + 1 отрезок.
2. Четности a и b совпали, при этом $b \leqslant a$. Тогда мы можем вообще не создавать новые отрезки, а просто продлить старые на один ряд.
3. Четности a и b совпали, но $b > a$. Тогда нам понадобится провести $b - a$ новых отрезков, а оставшиеся мы можем продлить.

Таким образом, можно посчитать стоимость каждого следующего ряда. А именно, если ввести прошлую функцию как $f(a, b)$, то нам понадобится $a + 1 + f(a, b) \cdot k + f(b, a) \cdot (k - 1) + f(b, n)$ вертикальных отрезков. Так мы получили решение сложностью $O(1)$.

Задача B. Альтернативные правила

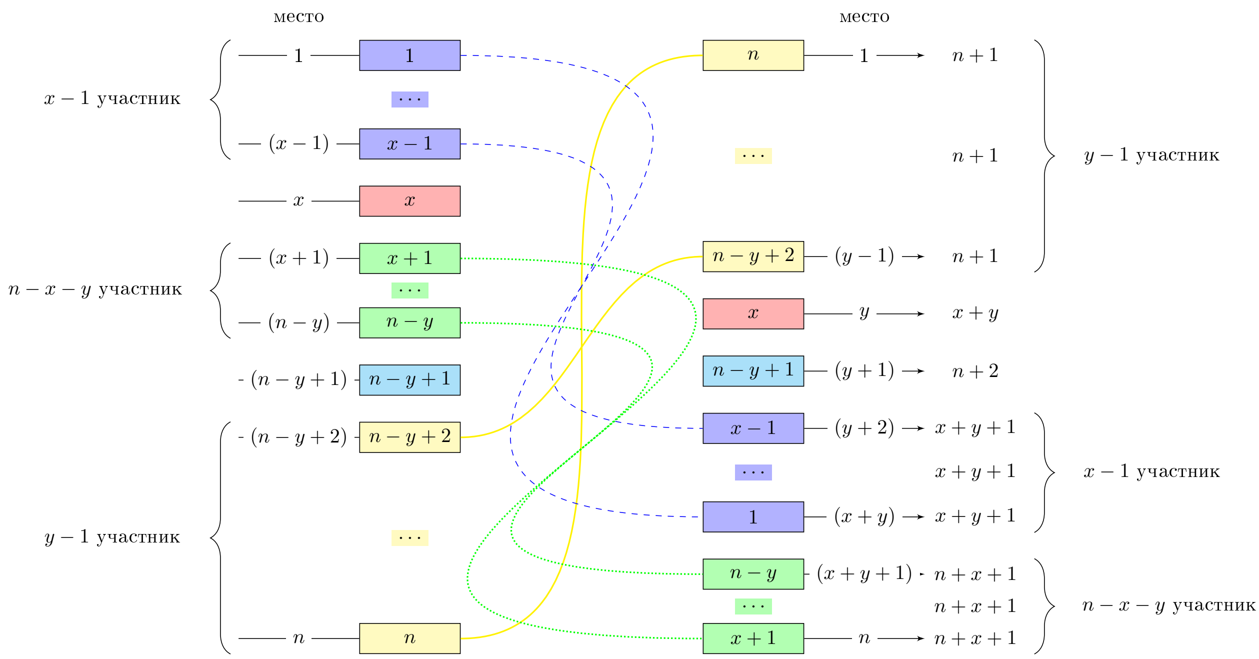
Не умаляя общности предположим, что $x \leqslant y$. Для удобства занумеруем участников от 1 до n в том порядке, какие места они заняли в первом туре. Таким образом интересующий нас участник – это участник x.

Докажем формулу: $\mathrm{MIN\_PLACE} = \max(1, \min(n, x + y - n + 1))$

1. Первый случай: $x + y < n$. Покажем как можно добиться первого места в олимпиаде. Для этого построим следующий пример:

   - i-й участник $(1 \leqslant i \leqslant x - 1)$ занимает $(x + y + 1 - i)$-е место во втором туре (сумма — $x + y + 1$)
   - x-й участник занимает y-е место во втором туре (сумма — $x + y$)
   - j-й участник $(x + 1 \leqslant j \leqslant n - y)$ занимает $(n + x + 1 - j)$-е место во втором туре (сумма — $n + x + 1$)
   - $(n - y + 1)$-й участник занимает $(y + 1)$-е место во втором туре (сумма — $n + 2$)
   - t-й участник $(n - y + 2 \leqslant t \leqslant n)$ занимает $(n + 1 - t)$-е место во втором туре (сумма — $n + 1$)

   Иллюстрация ниже поясняет данный пример

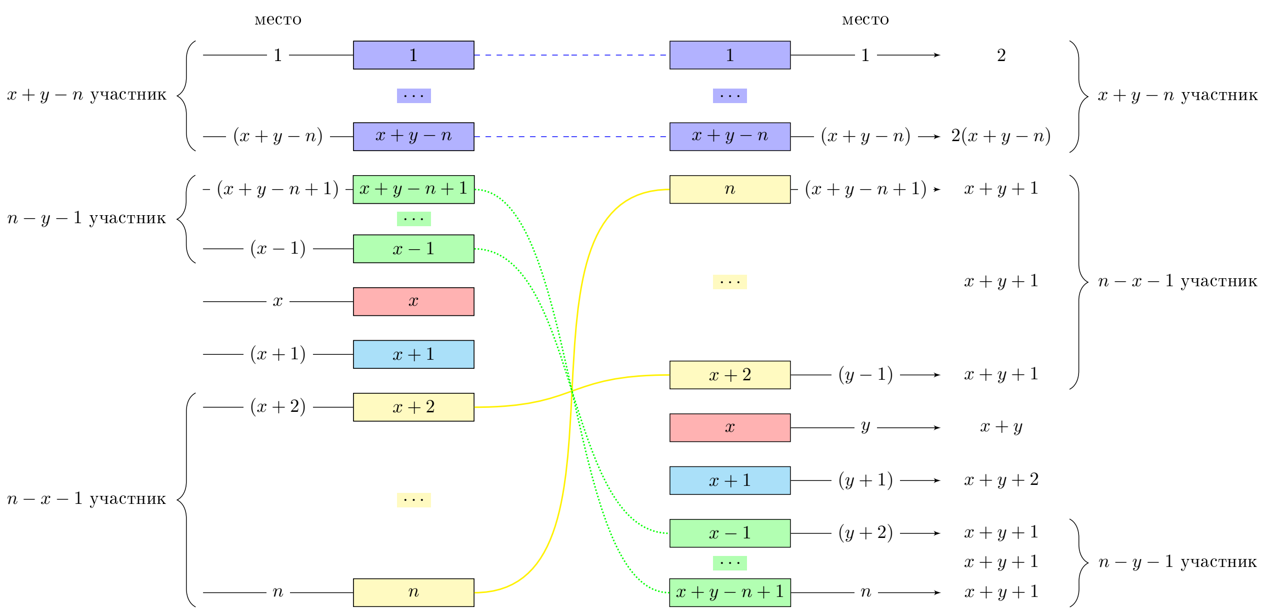


2. Второй случай: $x+y \geqslant n+1$; $y \neq n$

   Тогда участник с номером k ($k \leqslant x+y-n < x$) получит не более чем $x+y-n+n = x+y$ в сумме, то есть гарантированно обгонит главного героя. То есть, мы не можем занять места лучше, чем $x+y-n+1$. Для такой оценки ниже приведём пример:

   - i-й участник ($1 \leqslant i \leqslant x+y-n$) занимает i-е место во втором туре (сумма — $2i < x+y$)
   - j-й участник ($x+y-n+1 \leqslant j \leqslant x-1$) занимает $(x+y+1-j)$-е место во втором туре (сумма — $x+y+1$)
   - x-й участник занимает y-е место во втором туре (сумма — $x+y$)
   - $(x+1)$-й участник занимает $(y+1)$-е место во втором туре (сумма — $x+y+2$)
   - t-й участник ($x+2 \leqslant t \leqslant n$) занимает $(x+y+1-t)$-е место во втором туре (сумма — $x+y+1$)

   Иллюстрация ниже поясняет данный пример



3. Третий случай: $x + y \geqslant n + 1$; $y = n$

   Тогда участник с номером k ($k \leqslant x + y - n + 1 = x + 1$) получит не более чем $x + y - n + 1 + n - 1 = x + y$ в сумме, то есть гарантированно обгонит главного героя. То есть, мы не можем занять места лучше, чем $x + y - n + 1$. Для такой оценки ниже приведём пример:

   - Участник $i < x$ занимает x-е место, обгоняя $x + y$
   - Участник $x + 1$ занимает $(n - 1)$-е место, обгоняя $x + y$
   - Участник j $(x + 2 \leqslant j \leqslant n)$ занимает $(x + y + 1 - j)$-е место

   Отдельный случай: x = y = n, тогда исход очевиден

Формула для минимального места доказана. Докажем аналогично формулу для максимального места:

Докажем формулу: $\text{MAX\_PLACE} = \min(n, x + y - 1)$

1. Первый случай: $x + y \geqslant n + 1$. Тогда можно привести такой пример, при котором мы займём последнее место:

   - i-й участник $(1 \leqslant i \leqslant x + y - n - 1)$ занимает $(y + x - n - i)$-е место во втором туре (сумма — $y + x - n$)
   - j-й участник $(x + y - n - 1 \leqslant j \leqslant n)$ занимает $(x + y - j)$-е место во втором туре (сумма — $y + x$)

2. Второй случай: $x + y \leqslant n$ Пусть участник с номером k $(x + y \leqslant k)$ во втором туре займёт не менее чем $1 + x + y > x + y$, то есть всё равно не обгонит $x + y$ Значит, герой займёт в худшем случае не более чем $x + y - 1$ место:

   - i-й участник $(1 \leqslant i \leqslant x + y - 1)$ занимает $(y + x - i)$-е место во втором туре (сумма — $y + x - n$)
   - j-й участник $(x + y - 1 \leqslant j \leqslant n)$ занимает $(x + y + n - j)$-е место во втором туре (сумма — $y + x$)

Таким образом, задача свелась к задаче вывод двух чисел – $\langle \max(1, \min(n, x + y - n + 1)),\ \ \min(n, x + y - 1) \rangle$

Задача C. Деление

Возможно несколько случаев, которые надо учесть в этой задаче.

1. $m > b^k$, то есть какое-бы число не загадала Маша, оно меньше m. Тогда если $n > 1$, то надо спросить про 0 позиций, так как загаданное число больше 0 и меньше m, значит точно не делится на m. Если $n = 1$, то надо спросить про 1 позицию, так как число может быть равно 0 и делится, или равно 1 и не делится.

2. Не 1 случай и существует такое целое $k < n$, что b^k делится на m. Тогда про все позиции после k-й с конца спрашивать нет смысла, а про последние k надо узнать (так как если последние k цифр равны 0, то в этом случае число точно делится, а если хотя-бы одна из них равна 1, то деления уже не будет, значит чтобы определить случай последних k цифр 0 надо про все k последних цифр спросить).

3. Не 1 случай, не 2 случай, $b = 2$ и $n > 1$. В этом случае про первую цифру мы знаем что она не 0, значит она равна 1. Про остальные цифры мы не знаем чему они равны, и изменение любой цифры из числа, делящегося на m сделает число, не делящееся на m, значит чтобы убедится, что число делится на m, надо спросить про все цифры кроме 1.

4. Все остальные случаи. Про все цифры мы не знаем чему они равны, и изменение любой цифры из числа, делящегося на m сделает число, не делящееся на m, значит чтобы убедится, что число делится на m, надо спросить про все цифры.

Задача D. Небоскрёбы

Пусть мы решаем задачу на каком-то массиве m длины n. Найдем в этом массиве минимальный элемент. Пусть он находится на позиции i $(1 \leqslant i \leqslant n)$. Мы можем сделать небоскрёб на участке i максимально высоким, присваивая $a_i = m_i$.

Теперь нам необходимо сделать выбор — нам нужно приравнять к a_i либо левую часть массива $(a_1 = a_i, a_2 = a_i, \ldots, a_{i-1} = a_i)$, либо правую часть массива $(a_{i+1} = a_i, a_{i+2} = a_i, \ldots, a_n = a_i)$, и решать рекурсивно задачу для оставшейся части массива, пока не придем к массиву единичной длины.

Вышеописанная рекурсивная задача имеет всего n различных состояний. В зависимости от метода поиска минимального элемента на отрезке можно получить решение сложности $O(n^2)$, $O(n\sqrt{n})$ или $O(n \log n)$.

Есть и другое решение. Можно доказать, что все ответы выглядят следующим образом: с начала массива высота небоскрёбов неубывает, затем начиная с какого-то небоскрёба высота невозрастает. Назовем «вершиной» небоскрёб, на котором происходит смена направления роста величины небоскрёбов.

Можно сделать два прохода. Построим два массива l и r длины n. В первом проходе мы идем слева направо. Пусть мы сейчас на позиции i. Если m_i — наименьший элемент среди $m_1, \ldots, m_i$, тогда $l_i = i \times m_i$. Иначе, возьмем среди чисел $m_1, m_2, \ldots, m_{i-1}$ самое правое число, меньшее m_i. Пусть позиция этого числа j, тогда $l_i = l_j + (i-j) \times m_i$. Аналогично построим массив r. «Вершиной» является такой небоскрёб t, что значение $l_t + r_t - m_t$ максимально.

Сложность этого решения может быть $O(n^2)$, $O(n \log n)$, $O(n)$ в зависимости от способа поиска «ближайших» чисел слева и справа, меньших текущего.

Задача E. Новый год

Применим метод сканирующем прямой. Добавим события добавления отрезка и события удаления.

Будем поддерживать динамику $dp_{i,mask}$, где i это номер последнего события, которое было обработано, и $mask$ это маска, в которой k бит, которая как-то сопоставляет биты отрезкам, которые пересекают сканирующую прямую. 1 в маске означает, что отрезок, который сопоставлен данном биту, взят.

Тогда, как перейти от одной координаты к другой? Для каждой маски можем подсчитать число бит, и если оно нечетное, то добавим дистанцию между двумя точками.

Как добавить отрезок? Так как в точке пересекается не более k отрезков, то при добавлении есть какой-то свободный бит. Сопоставим этот бит данном отрезку и обновим некоторые маски.

Удаление делается аналогично. Мы освобождаем сопоставленный бит и обновляем какие-то маски.

Так же можно заметить что для пересчета i-го слоя нужен только $(i-1)$-й слой, поэтому нам нужно только $O(2^k)$ дополнительной памяти.

Итоговая асимптотика $O(n \log n + n2^k)$.

Задача F. Конкатенация с пересечением

Для всех $1 \leqslant i \leqslant n$ обозначим за fa_i длину наибольшего общего префикса строк $a[i, n]$ и $s[1, m-1]$, за fb_i длину наибольшего общего суффикса строк $b[1, i]$ и $s[2, m]$. Значения fa легко получаются из z-функции строки «$s\#a$», значения fb из z-функции строки «$\overline{s}\#\overline{b}$» (тут за $\overline{s}$ обозначается перевернутся строка s).

Зафиксируем l_1 и r_2. Заметим, что поскольку отрезки пересекаются, то $l_1 \leqslant r_2$. Также сумма длин отрезков равна m и они пересекаются, поэтому $r_2 \leqslant l_1 + m - 2$. Легко видеть, что отрезки будут пересекаться, тогда и только тогда, когда $l_1 \leqslant r_2 \leqslant l_1 + m - 2$.

Заметим, что если для фиксированных l_1 и r_2 описанные неравенства верны, то количество пар отрезков с такими l_1 и r_2 равно $\max(fa_{l_1} + fb_{r_2} - m + 1, 0)$. Поэтому ответ на задачу равен

$$\sum_{1 \leqslant l_1 \leqslant r_2 \leqslant \min(l_1+m-2,n)} \max(fa_{l_1} + fb_{r_2} - m + 1, 0) = \sum_{l_1=1}^{n} \sum_{r_2=l_1}^{\min(l_1+m-2,n)} \max(fa_{l_1} + fb_{r_2} - m + 1, 0).$$

Сделаем два дерева Фенвика на массивах размера m. Будем перебирать l_1 от 1 до n. Для $r_2 \in [l_1, \min(l_1 + m - 2, n)]$ будем в позиции $m - 1 - fb_{r_2}$ прибавлять 1, в позиции $m - 1 - fb_{r_2}$ второго дерева прибавлять fb_{r_2}. После этого для фиксированного l_1 легко найти сумму из сумм в дереве Фенвика на префиксах $i \leqslant fa_{l_1}$.

Получаем решение за $O(n \log n)$.