Разбор олимпиады 2025

Задача 1. Плюс-минус

Пусть дана интегрируемая функция $f\colon \mathbb{R} \to \mathbb{R}$ такая, что $\int_{-\infty}^{+\infty} f(x)\,dx = 2025$. Найдите $\int_{-\infty}^{+\infty} f\left(x - \frac{1}{x}\right)\,dx$.

Ответ должен быть представлен в максимально простой форме, т.е. не содержать знаков суммы, произведения, троеточий.

Ответ. 2025.

Решение 1.

$$\int_{-\infty}^{+\infty} f\left(x-\frac{1}{x}\right)\,dx = \int_{0}^{+\infty} f\left(x-\frac{1}{x}\right)\,dx + \int_{0}^{+\infty} f\left(-x+\frac{1}{x}\right)\,dx =$$

$$= \int_{-\infty}^{+\infty} f\left(2\operatorname{sh} y\right) e^{y}\,dy + \int_{-\infty}^{+\infty} f\left(2\operatorname{sh} y\right) e^{-y}\,dy = \int_{-\infty}^{+\infty} f\left(2\operatorname{sh} y\right)\cdot 2\operatorname{ch} y\,dy = \int_{-\infty}^{+\infty} f\left(x\right)\,dx = 2025.$$

Решение 2. Рассмотрим $\varphi(x) = x - \frac{1}{x}$. На промежутках $I_1 = (-\infty; 0)$ и $I_2 = (0; +\infty)$ функция строго возрастает и дифференцируема с положительной производной, причём $\lim_{x\to+\infty}\varphi(x) = \lim_{x\to 0-}\varphi(x) = +\infty$ и $\lim_{x\to-\infty}\varphi(x) = \lim_{x\to 0+}\varphi(x) = -\infty$. Значит, функции $\varphi_1(x)$ и $\varphi_2(x)$, определяемые как ограничения $\varphi(x)$ на I_1 и I_2 соответственно, осуществляют диффеоморфизм между своими областями определения и $\mathbb{R}$. По теореме о замене переменных

$$\int_{-\infty}^{+\infty} f\left(x-\frac{1}{x}\right)\,dx = \int_{I_1} f\left(\varphi_1(x)\right)\,dx + \int_{I_2} f\left(\varphi_2(x)\right)\,dx =$$

$$= \int_{\mathbb{R}} |\varphi_1^{-1}(y)'|\cdot f\left(y\right)\,dy + \int_{\mathbb{R}} |\varphi_2^{-1}(y)'|\cdot f\left(y\right)\,dy = \int_{\mathbb{R}} (|\varphi_1^{-1}(y)'| + |\varphi_2^{-1}(y)'|)\cdot f\left(y\right)\,dy.$$

Во-первых, так как φ_1, φ_2 строго возрастают, то таковы и $\varphi_1^{-1}, \varphi_2^{-1}$, поэтому модули в формуле выше можно снять. Во-вторых, $\varphi_1^{-1}(y)$ и $\varphi_2^{-1}(y)$ есть корни уравнения $x^2 - 1 = yx$, сумма которых равна y по теореме Виета, поэтому производная суммы равна 1. Таким образом, искомый интеграл равен интегралу $f(x)$ на прямой, то есть 2025.

Решение 3.

$$\int_{-\infty}^{+\infty} f\left(x-\frac{1}{x}\right)dx = \int_{-\infty}^{0} f\left(x-\frac{1}{x}\right)dx + \int_{0}^{+\infty} f\left(x-\frac{1}{x}\right)dx = \{y = -\frac{1}{x}\}$$

$$= \int_{0}^{+\infty} f\left(-\frac{1}{y}+y\right)\frac{1}{y^2}dy + \int_{-\infty}^{0} f\left(-\frac{1}{y}+y\right)\frac{1}{y^2}dy.$$

Сложим последние два выражения, получим

$$2\int_{-\infty}^{+\infty} f\left(x-\frac{1}{x}\right)dx = \int_{0}^{+\infty} f\left(y-\frac{1}{y}\right)\left(1+\frac{1}{y^2}\right)dy + \int_{-\infty}^{0} f\left(y-\frac{1}{y}\right)\left(1+\frac{1}{y^2}\right)dy =$$

$$= 2\int_{-\infty}^{+\infty} f\left(x\right)\,dx,$$

откуда вновь находим, что искомый интеграл равен 2025.

Задача 2. Катан

На острове Катан расположено 100 городов, но пока нет ни одной дороги. Властелин Катана, король Ричард Душный, издал указ построить на острове такую систему дорог, чтобы из каждого города выходило нечётное число дорог, а общее количество дорог на острове было чётным. При этом дорога соединяет 2 города, и между каждыми двумя городами может быть не более одной дороги. Сколько существует способов построить такую систему дорог, чтобы удовлетворить желание короля?

Ответ должен быть представлен в максимально простой форме, т.е. не содержать знаков суммы, произведения, троеточий.

Ответ. $2^{C_{99}^2-1}$.

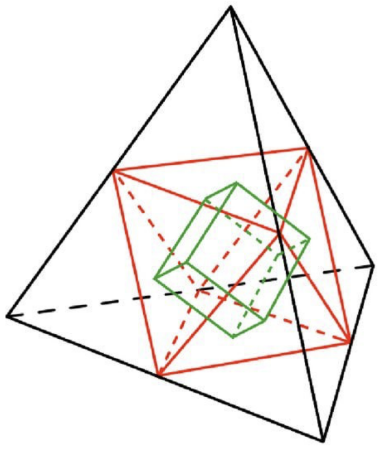
Решение. Выберем произвольную вершину, назовем ее v, и удалим. На оставшихся 99 вершинах проведем произвольно ребра, но так, чтобы их было четное число. Всего вариантов провести ребра $2^{C_{99}^2}$, а вариантов провести четное число ребер тогда $2^{C_{99}^2-1}$. Заметим, что количество вершин нечетной степени окажется четным по лемме о рукопожатиях.

Теперь вернем вершину v и проведем от нее ребра к тем и только тем вершинам, у которых была нечетная степень. Таких вершин было четное число, значит, мы проведем еще четное число ребер, итого в графе снова будет четное число ребер. При этом у всех 100 вершин окажется четная степень. Заметим, что каждый граф на 100 вершинах с четным числом ребер и четной степенью всех вершин ровно одним способом мог быть получен нашими действиями. Значит, всего таких графов ровно $2^{C_{99}^2-1}$.

Теперь инвертируем в графе все ребра. Поскольку максимум ребер могло быть $C_{100}^2 = 50 \cdot 99$ – четное число, и у нас было ребер четное число, то после инвертирования ребер их снова будет четное число. А вот степени всех вершин, наоборот, поменяют четность. То есть мы получим граф со всеми нечетными степенями вершин и с четным количеством ребер. Опять же заметим, что каждый такой граф мог быть получен ровно одним способом. Значит, искомых графов столько же, а именно $2^{C_{99}^2-1}$.

Задача 3. Матрёшка

Куб с длиной ребра 1 вписан в тетраэдр, как показано на рисунке (грани красного октаэдра соединяют середины рёбер тетраэдра, и вершины куба – это середины граней октаэдра).



Найдите количество различных прямых, проходящих через центр куба, таких что сумма квадратов расстояний от вершин октаэдра, тетраэдра и куба до данной прямой максимальна.

В качестве ответа напишите целое положительное число, или -1, если ответа не существует.

Ответ. -1.

Решение. Пусть O — центр куба, A — точка с координатами $(a,\ b,\ c)$, P — точка с координатами $(x,\ y,\ z)$, H — основание перпендикуляра из A на прямую OP. Рассмотрим величину $|AH|^2|OP|^2$:

$$|AH|^2|OP|^2 = (|OA|^2 - |OH|^2)|OP|^2 = |OA|^2|OP|^2 - (\overline{OA} \cdot \overline{OP})^2 =$$
$$= (a^2 + b^2 + c^2)(x^2 + y^2 + z^2) - (ax + by + cz)^2.$$

Таким образом, данная величина задает квадратичную форму по x, y и z, поэтому и сумма таких величин по всем точкам, участвующим в задаче, задает квадратичную форму. Квадратичная форма диагонализуется в ортонормированном базисе; кроме того, поскольку она положительно определена, ее линии уровня будут эллипсоидами. Однако можно заметить, что наша форма инвариантна при повороте на 120°

вокруг четырёх осей, проходящих через вершину тетраэдра и середину противоположной стороны; отсюда следует, что линии уровня нашей формы на самом деле сферы. Теперь, чтобы получить величину, необходимую в задаче, достаточно рассмотреть все точки P на расстоянии 1 от O, но это будет линией уровня формы; таким образом, искомая величина не зависит от прямой.

Задача 4. Инклюзивный марьяж

Инклюзивным марьяжем назовём комбинацию из двух карт: короля и дамы, причём их масти могут быть произвольными. Стандартная колода состоит из 52 карт, среди которых ровно по 4 короля и дамы. Стопку хорошо перемешали и начали по одной (без возвращения) вынимать оттуда карты. Найдите математическое ожидание числа карт, которое нужно вытянуть, чтобы получить инклюзивный марьяж. Ответ представить в виде несократимой дроби.

Ответ. $\dfrac{689}{45}$.

Решение. Для решения нам понадобится следующая лемма:

Лемма. При вытаскивании карт из колоды размера n, в которой помечено $k \leqslant n$ карт, математическое ожидание числа карт, которое нужно вытянуть, чтобы достать помеченную, равно $\frac{n+1}{k+1}$.

Доказательство 1. Пусть ξ — момент появления первой помеченной карты. Событие $\{\xi > m\}$ означает, что среди первых m карт нет помеченных, вероятность чего равна $\frac{C_{n-m}^k}{C_n^k}$ (среди всех C_n^k расстановок выделенных карт нам подходят только те, где карты выбираются среди последних $n-m$). Поэтому

$$\mathsf{E}\xi = \sum_{m=1}^{n-k+1} m \cdot \mathsf{P}(\xi = m) = \sum_{m=1}^{n-k+1} \mathsf{P}(\xi \geqslant m) = \sum_{m=0}^{n-k} \mathsf{P}(\xi > m) =$$

$$= \frac{1}{C_n^k} \sum_{m=0}^{n-k} C_{n-m}^k = \frac{C_{n+1}^{k+1}}{C_n^k} = \frac{n+1}{k+1},$$

где в предпоследнем равенстве мы воспользовались известным тождеством о сумме чисел на диагонали треугольника Паскаля ($\sum\limits_{j=k}^{n} C_j^k = C_{n+1}^{k+1}$).

Доказательство 2. Пусть a_n — искомое математическое ожидание для колоды из n карт (число помеченных карт k фиксировано). По формуле полного математического ожидания:

$$a_n = \frac{k}{n} \cdot 1 + \frac{n-k}{n} \cdot (1 + a_{n-1}).$$

По индукции легко понять, что под эту реккуренту подходит искомый ответ.
Для базы индукции ($n = k$, т.е. все карты помеченные) это очевидно.
Переход индукции тривиален:

пусть для $n-1$ верно: $a_{n-1} = \frac{(n-1)+1}{k+1} = \frac{n}{k+1}$.

Покажем, что верно для n:

$$a_n = \frac{k}{n} + \left(1 + \frac{n}{k+1}\right)\frac{n-k}{n} = 1 + \frac{n-k}{k+1} = \frac{n+1}{k+1}.$$

Следовательно, по индукции для всех $n \geq k$ верно: $a_n = \frac{n+1}{k+1}$.

Доказательство 3. Добавим на верх колоды ещё одну помеченную карту и «закольцуем» колоду — расположим $n+1$ карту по кругу в том порядке, в котором они лежат в колоде. Случайно «повернём» наш круг (то есть будем считать первой картой не нашу добавленную, а случайную из $(n+1)$-ой). Несложно понять, что распределение помеченных карт равномерное, и в силу симметрии математическое ожидание числа карт после произвольной фиксированной помеченной карты равно $\frac{n+1-(k+1)}{k+1}$. Значит, искомое математическое ожидание равно числу карт после добавленной плюс одна помеченная карта, которую мы достанем в самом конце — итого $\frac{n+1-(k+1)}{k+1} + 1 = \frac{n+1}{k+1}$, что и требовалось.

Вернёмся к задаче. Решим её в общем случае, для m мастей и n номиналов (итого, nm карт; в нашем случае $n = 13, m = 4$). Пусть $\xi \in \{1, \ldots, nm\}$ — номер первой карты в колоде, которая либо король, либо дама. По лемме $\mathsf{E}\xi = \frac{nm+1}{2m+1}$ — в колоде из nm карт пометили m королей и m дам. При фиксированном ξ распределение остальных $2m-1$ помеченных карт такое же равномерное. Нам осталось найти даму (если изначально взят король) или короля (если изначально взята дама). Их осталось ровно m (так как ранее мы их не вынимали). Таким образом, задача опять свелась к лемме — среди оставшихся $nm - \xi$ карт пометили m штук. Количество карт η, которое нужно вынуть до первой помеченной, имеет условное математическое ожидание при условии ξ, равное $\frac{nm-\xi+1}{m+1}$. Итого, по формуле полного математического ожидания:

$$\mathsf{E}(\xi+\eta) = \mathsf{E}\xi + \mathsf{E}(\mathsf{E}(\eta|\xi)) = \mathsf{E}\left(\xi + \frac{nm-\xi+1}{m+1}\right) = \frac{nm+1}{m+1} + \frac{m}{m+1}\cdot\mathsf{E}\xi =$$

$$= \frac{nm+1}{m+1} + \frac{m}{m+1}\cdot\frac{nm+1}{2m+1} = \frac{(nm+1)(3m+1)}{(m+1)(2m+1)}.$$

Для $n = 13, m = 4$ ответ равен $\frac{689}{45}$.

Замечание. Участники олимпиады предложили не менее красивые решения, использующие лемму выше. Рассмотрим одно из них.

Пусть ξ — момент появления первого короля, а η — первой дамы. Нас просят найти $\mathsf{E}\max\{\xi,\eta\}$. Максимум можно связать с минимумом следующим образом: $\max\{\xi,\eta\}+\min\{\xi,\eta\} = \xi+\eta$. Но $\min\{\xi,\eta\}$ — момент появления первой из восьми выделенных карт: дамы или короля, поэтому его математическое ожидание можно найти с помощью леммы выше для $n = 52$ и $k = 8$. Таким образом,

$$\mathsf{E}\max\{\xi,\eta\} = \mathsf{E}\xi + \mathsf{E}\eta - \mathsf{E}\min\{\xi,\eta\} = \frac{52+1}{4+1} + \frac{52+1}{4+1} - \frac{52+1}{8+1} = \frac{689}{45}.$$

Прелесть данного решения заключается в том, что оно работает и в случае различного количества королей и дам, в то время как авторское решение требует дополнительных уточнений.

Задача 5. ЫЫЫ

Пусть подпространства $V, U \subseteq \mathbb{R}^4$ заданы следующим образом:

$$V = \langle v_1, v_2, v_3\rangle, \quad U = \{y \in \mathbb{R}^4 \mid Ы\, y = 0\},$$

где

$$v_1 = \begin{pmatrix} 3 \\ -2 \\ 2 \\ 1 \end{pmatrix},\ v_2 = \begin{pmatrix} -1 \\ 2 \\ -2 \\ 3 \end{pmatrix},\ v_3 = \begin{pmatrix} 3 \\ -3 \\ -1 \\ 2 \end{pmatrix}, \quad Ы = \begin{pmatrix} 2 & -1 & -3 & -2 \end{pmatrix}.$$

Пусть множество матриц определено следующим образом:

$$F = \{A \in M_4(\mathbb{R}) \mid \text{столбцы } A \text{ лежат в } V \text{ и строки } A \text{ лежат в } U\}.$$

Найти размерность

$$W = \{A \in M_4(\mathbb{R}) \mid A = A^T,\ A \in F\}.$$

В качестве ответа напишите целое число, или -1, если это не линейное пространство.

Ответ. 3.

Решение. Ясно, что в этом случае матрицы из W имеют строки и столбцы в подпространстве $V \cap U$. Найдём размерность этого подпространства. Поставим векторы v_i по столбцам матрицы $B = (v_1|v_2|v_3)$. Тогда пересечение можно найти так. В начале решаем систему:

$$\begin{cases} v = B\alpha \\ Ыv = 0 \end{cases} \Leftrightarrow \begin{cases} v = B\alpha \\ ЫB\alpha = 0 \end{cases}$$

Тогда на $\alpha \in \mathbb{R}^3$ получаем уравнение

$$-4\alpha_2 + 8\alpha_3 = 0 \Leftrightarrow \alpha_2 - 2\alpha_3$$

Значит, ФСР будет

$$s_1 = \begin{pmatrix} 0 \\ 2 \\ 1 \end{pmatrix},\ s_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Тогда порождающие для пересечения будут

$$w_1 = Bs_1 = \begin{pmatrix} 1 \\ 1 \\ -5 \\ 8 \end{pmatrix},\ w_2 = Bs_2 = \begin{pmatrix} 3 \\ -2 \\ 2 \\ 1 \end{pmatrix}$$

Значит, пересечение двумерно.

Пусть $R = (w_1|w_2)$ – базис пересечения $V \cap U$, сложенный в матрицу R. Тогда рассмотрим матрицу

$$A = R\begin{pmatrix} a & b \\ c & d \end{pmatrix} R^T = aw_1w_1^T + bw_1w_2^T + cw_2w_1^T + dw_2w_2^T.$$

Заметим, что матрицы $w_iw_j^T$ являются базисом пространства матриц, у которых строки и столбцы лежат в $V \cap U$, поскольку можно напрямую проверить, что они линейно независимы, а далее дополнить w_1 и w_2 до базиса и посмотреть, что в нем матрица будет выглядеть как левый верхний минор. Значит, вид матрицы выше — это общий вид матрицы из такого подпространства. Теперь надо лишь добавить условие $A = A^T$. Получится условие

$$R\begin{pmatrix} a & b \\ c & d \end{pmatrix} R^T = R\begin{pmatrix} a & c \\ b & d \end{pmatrix} R^T$$

Это в свою очередь равносильно условию

$$R\begin{pmatrix} & b-c \\ c-b & \end{pmatrix} R^T = 0.$$

Так как матрица R с линейно независимыми столбцами, то она обратима слева. Соответственно, матрица R^T обратима справа. Это значит, что условие превращается в

$$\begin{pmatrix} & b-c \\ c-b & \end{pmatrix} = 0$$

А это значит, что у нас только одно уравнение на переменные a, b, c, d, и искомая размерность будет равна $4 - 1 = 3$.

Задача 6. Ширма

Предприниматель Игорь Мясников наконец-то познал успешный успех, и теперь он каждый день получает стабильный и постоянный доход, равный 1 млн. рублей. Но вдруг злая ведьма Инфляция наложила на него проклятие, которое заключается в следующем. Каждый день с момента наложения проклятия текущая зарплата Игоря может случайно уменьшиться в 2 раза, причём происходит это независимо от прошлого и в n-ый день ($n \geqslant 2$) с вероятностью $\frac{2}{n}$. Помогите Игорю: найдите вероятность того, что даже при таком бедствии он сможет и дальше неограниченно обогащаться или, проще говоря, сумма его ежедневных доходов будет стремиться к бесконечности.

Ответ. 0.

Решение. Пусть $\xi_1, \xi_2, \ldots$ — независимые случайные величины, причём ξ_i равна 1, если в i-ый день произошло уменьшение зарплаты, и 0 иначе. Тогда $\xi_n \sim \mathrm{Bern}(\frac{2}{n})$. Зафиксируем некоторую константу $a > 1$, значение которой мы подберём позднее, и рассмотрим отрезок $S_n = [[a^n]; [a^{n+1}]]$. Рассмотрим вероятность события A_n, что на S_n не произошло ни одного уменьшения зарплаты:

$$\mathsf{P}(A_n) = \prod_{k=[a^n]}^{[a^{n+1}]} \left(1 - \frac{2}{k}\right) = \exp\left[\sum_{k=[a^n]}^{[a^{n+1}]} \ln\left(1 - \frac{2}{k}\right)\right] \leqslant$$

$$\leqslant \exp\left[-\sum_{k=[a^n]}^{[a^{n+1}]} \frac{2}{k}\right] = \exp\left[-\int_{a^n}^{a^{n+1}} \frac{2}{x}\,dx + o(1)\right] = \exp\left[-2\ln x\Big|_{a^n}^{a^{n+1}} + o(1)\right] =$$

$$= \exp\left[-2\ln a\right] \cdot (1 + o(1)) \sim \frac{1}{a^2}.$$

Таким образом, начиная с некоторого n_0, для всех $n \geqslant n_0$ справедлива оценка $\mathsf{P}(A_n) \leqslant a^{-3/2}$.

Рассмотрим несколько иной ряд

$$10^6 \cdot \sum_{n=1}^{\infty} \frac{a^{X_n}}{2^n}, \quad X_n = \eta_1 + \ldots + \eta_n$$

где η_i — независимые с.в. с распределением $\mathrm{Geom}(1 - a^{-3/2})$. Так как на каждом отрезке S_n с достаточном большим номером n вероятность хотя бы одного уменьшения не меньше $1 - a^{-3/2}$, то исходный ряд можно мажорировать новым. С другой стороны, для нового ряда можно использовать радикальный признак Коши:

$$\sqrt[n]{\frac{a^{X_n}}{2^n}} = \frac{a^{X_n/n}}{2} \xrightarrow{\text{п.н.}} \frac{a^{\mathsf{E}\eta_1}}{2},$$

где $\mathsf{E}\eta_1 = \frac{1}{1-a^{-3/2}}$. Рассмотрим полученное выражение при $a \to 1+0$:

$$\frac{a^{\mathsf{E}\eta_1}}{2} = \frac{1}{2}\exp\left(\frac{a^{3/2}}{a^{3/2}-1}\cdot \ln a\right) = \frac{1}{2}\exp\left(\frac{a^{3/2}\cdot\frac{2}{3}}{a^{3/2}-1}\cdot \ln a^{3/2}\right) = \frac{1}{2}\exp\left(a^{3/2}\cdot\frac{2}{3}(1+o(1))\right) \to \frac{1}{2}e^{2/3} < 1,$$

поэтому при выборе значения a, близкого к единице, ряд выше сходится почти наверное, а значит, и изначальный ряд сходится с вероятностью 1, то есть ответ на задачу — 0.

Замечание. Как можно заметить, решающим фактором сходимости ряда стало соотношение на две константы: C, множитель у $1/n$ из вероятности уменьшения в n-ый день, и k, во сколько раз уменьшается зарплата (в нашем случае $C = k = 2$). В общем случае наше рассуждение показывает, что при $e^{1/C} < k$ ряд почти наверное сходится.

Оказывается, в каком-то смысле верен и обратный факт: если $e^{1/C} > k$, то ряд почти наверное расходится. Данное утверждение требует чуть более тонкого анализа асимптотического поведения случайных величин вида $\xi_{[a^n]}+\xi_{[a^n]+1}+\ldots+\xi_{[a^{n+1}]}$. В решении выше найден лишь предел вероятности того, что данная сумма равна 0, хотя в действительности данная сумма и вовсе сходится по распределению к $\mathrm{Pois}(2\ln a)$. Это можно показать, сославшись, например, на предельную теорему Пуассона.

Задача 7. Игра с монеткой

Алексей в моменты ожидания чего-либо любит подбрасывать монетку. Как начинающий статистик он решил выписать результаты последних N бросков монетки: вместо «орла» он использовал ноль, а вместо «решки» — единицу. Помогите ему найти подотрезок длины хотя бы K с минимальной дисперсией результатов.

Броски монетки можно считать независимыми реализациями случайной величины из распределения $Bern(0.5)$: вероятность выпадения «орла» равна вероятности выпадения «решки», они обе равны 0.5.

Формат ввода: В первой строке идут два числа: N $(2 \le N \le 10^5)$ и K $(1 \le K \le N)$ — число бросков в записях Алексея и минимальная длина подотрезка, интересующие Алексея, соответственно.

В второй строке даны результаты испытаний в виде строки S длины N из нулей и единиц — результаты бросков монетки.

Формат вывода: Выведите единственное число — минимально возможное значение дисперсии на подотрезке длины хотя бы K. Ответ будет засчитан, если каждый ответ по относительной или абсолютной погрешности отличается от эталона не более чем на 10^{-6}.

Решение. Переформулируем задачу. Дан массив из нулей и единиц длины N. Надо найти подотрезок длины хотя бы K с минимальной выборочной дисперсией.

Пусть p — значение среднего на подотрезке, тогда выборочная дисперсия равна $p(1-p)$, $p \in [0,1]$. Эта функция монотонно возрастает на $[0, 0.5]$ и убывает на $[0.5, 1]$, при этом значения в нуле и единице равны нулю. Тогда задача раскладывается на две похожие и независимые: найти подотрезок длины хотя бы K с минимальным/максимальным средним и выбрать лучший из ответов.

Рассмотрим задачу максимизации среднего, задача минимизации сводится к максимизации путем инвертирования значений массива в силу симметрии функции относительно максимума в точке 0.5.

Будем вещественнозначным бинарным поиском по ответу подбирать максимальное среднее p, что для него все еще существует подотрезок длины хотя бы K с средним хотя бы p. Данный подход тут применим, так $\forall p \le p^*$ такой подотрезок существует, а $\forall p > p^*$ уже нет. Теперь остается проверить существование подотрезка с средним $\ge p$ длины хотя бы K.

Вычтем из всех элементов текущее p, надо проверить существование отрезка с неотрицательным средним или суммой. Воспользуемся префиксными суммами $s_{i+1} = a_i + s_i$, $s_0 = 0$. Чтобы существовал подотрезок длины хотя бы K с неотрицательной суммой, заканчивающийся в позиции r, необходимо и достаточно выполнения условия

$$s_r - \min_{l \le i-k} s_l \ge 0$$

Для подсчета вычитаемого посчитаем префиксные минимумы $m_i = \min\limits_{j \le i-k} s_j$ по формуле $m_i = \min(m_{i-1}, s_{i-K})$, $i \ge K$. Таким образом, надо посчитать для каждой итерации массивы s, m и проверить, есть ли такой r, что $s_r \ge m_r$.

Отметим, что на самом деле можно явно не считать массив m и обновлять минимальное значение на префиксе по мере увеличения r.

Анализ потребляемых ресурсов

- Время работы.
  - Нужен вещественнозначный бинарный поиск, его итераций будет $O(\log \varepsilon^{-1})$, где ε — искомая точность ответа. В силу того, что рассматривают поиск по среднему, $\varepsilon = \frac{1}{N^2}$. Таким образом, получаем $O(\log N)$ итераций бинарного поиска.
  - Каждая итерация требует $O(N)$ времени на подсчет массивов s, m и проход по исходному массиву, чтобы вычесть текущую оценку на среднее.

  Таким образом, итоговое время работы составит $O(N \log N)$.

- Дополнительная память.
  - Бинпоиск не требует дополнительной памяти.
  - Итерация требует $O(N)$ доппамяти, чтобы посчитать массивы s, m. На самом деле можно сократить до $O(K)$ доппамяти, так как для расчета m_i достаточно помнить последние K значений $[s_{i-K}, \ldots, s_i]$, а их можно насчитывать в режиме онлайн и хранить в очереди.

Задача 8. Похожие геномы

Эвобиолог изучает вирусы, которые Homo Ludens способны найти в других мирах. Она закодировала их РНК некоторым кодом, являющимся натуральным числом. Ее интересуют, какие вирусы наиболее похожи, чтобы кластеризовать их, мерой похожести вирусов она выбрала XOR (взаимоисключающее или) чисел их РНК: чем меньше XOR, тем более похожи вирусы.

Вы, будучи первоклассным программистом, взялись помочь ей разработать базу данных вирусов, которая эффективно обрабатывает следующие запросы:

- `add G` — добавить в базу закодированную РНК нового вируса, где G — закодированная РНК;
- `get` — узнать пару наиболее похожих вирусов.

Изначально база эвобиолога пуста.

Формат ввода: На первой строке даны два числа N ($1 \le N \le 2 \cdot 10^5$) — число запросов к базе данных и длина генома вируса соответственно.

Далее в N строках идут запросы к базе данных.

- `1 G` — добавить в базу закодированную РНК нового вируса, G — натуральное число, ($0 \le G < 2^{30}$). Номер вируса соответствует порядковому номеру запроса этого типа среди всех.
- `2` — узнать пару наиболее похожих вирусов.

Гарантируется, что, все числа в запросах первого типа уникальны, на момент запроса второго типа есть хотя бы два вируса в базе.

Формат вывода: Для каждого запроса типа `get` выведите три числа: наименьший возможный XOR и номера наименее похожих вирусов в 1-индексации на очередной строке.

Если вариантов несколько, выведите любой. Номера в одной паре можно выводить в любом порядке.

Примечание: XOR двух чисел определяется следующим образом. Пусть $G_1 = 38_{10} = 100110_2$, $G_2 = 18_{10} = 010010_2$, тогда $G_3 = XOR(G_1, G_2) = 110100_2 = 52_{10}$: для i-го бита G_3 результат определяется следующим образом: $G_3[i] = G_1[i] \oplus G_2[i]$, где $\oplus$ — сложение по модулю 2.

Решение. Переформулируем задачу. Надо разработать структуру данных, хранящую множество целых чисел из полуинтервала $[0, 2^K)$ и способную обрабатывать следующие запросы:

- добавить число в множество;
- найти пару чисел a, b из множества с наименьшим `XOR`.

Будем хранить двоичные представления чисел в боре как строки длины K от старших разрядов к младшим. При этом будем хранить ответ и при добавлении очередного числа его обновлять.

Тогда запрос получения пары чисел из множества с наименьшим `XOR` сводится к запросу обновления ответа при добавлении.

Добавление числа X разбивается на две независимые стадии:

1. поиск для X в боре такого Y, что оно дает минимальный `XOR` с X, если результат меньше сохраненного, то обновляем ответ;

2. вставка X в бор.

Вторая операция классическая, тогда как первая уже нестандартная. Далее X не просто добавляемое число, а его битовая строка с старшими битами в начале. Будем спускать по бору как в классическом добавлении и при рассмотрении вершины на глубине d проверять, есть ли переход из нее по биту ~`X[i]`. При наличии такого переходим по нему, иначе по единственному имеющемуся. Так как глубины всех листьев равны K, переход для $d < K$ всегда есть хотя бы по одному биту.

Отдельно надо пояснить, почему такой «жадный» алгоритм работает. Рассмотрим переход на глубине d. Заметим, что $2^d > 2^{d-1} + \ldots + 2^0 = 2^d - 1$, а значит при общем префиксе $X[:d]$ переход по ~`X[i]` даст результат хотя бы $X[:d]_{10} \cdot 2^{d-1} + 2^d$, тогда как по `X[i]` даст результат не больший чем $X[:d]_{10} \cdot 2^{d-1} + 2^d - 1$, где $X[:d]_{10}$ это число, получаемое чтением первых d бит как двоичного представления некоторого числа.

Применяя это рассуждение рекурсивно, начиная с самого старшего бита, приходим к корректности вышеизложенного подхода.

Анализ потребляемых ресурсов

- Время работы.
  - Добавление числа требует двух спусков по бору высоты K. Это занимает $O(K)$ времени.
  - Получение ответа требует $O(1)$ времени, так как он обновляется в запросе добавления.

  Таким образом, итоговое время работы составит $O(NK)$.

- Дополнительная память.

  Бор требует $O(NK)$ дополнительной памяти в худшем случае.