Экзамен, 20 мая 2023

ШАД
классический трек

A. В одну строчку

Найдите предел последовательности $\{a_n\}_{n=1}^{\infty}$, где

$$a_n = \left|\sin\left(\pi\sqrt[3]{n^3+n^2}\right)\right|.$$

Ответ: $\frac{\sqrt{3}}{2}$

Решение

- $a_n = \left|\sin\left(\pi\sqrt[3]{n^3+n^2} - \pi n\right)\right|$, так как период функции $|\sin(t)|$ равен π.
- Посмотрим, чему равен предел функции внутри синуса.
  Рассмотрим функцию $f : \mathbb{R} \to \mathbb{R}$,
  где $f(x) = \pi\sqrt[3]{x^3+x^2} - \pi x$ и найдём её предел при $n \to +\infty$.
  Для этого воспользуемся разностью кубов:

$$f(x) = \pi\sqrt[3]{x^3+x^2} - \pi x = \pi\frac{x^2}{(x^3+x^2)^{\frac{2}{3}} + x\sqrt[3]{x^3+x^2} + x^2} \xrightarrow[x\to+\infty]{} \frac{\pi}{3}$$

- Ясно, что

$$\lim_{x\to+\infty}|\sin(f(x))| = \sin\left(\frac{\pi}{3}\right) = \frac{\sqrt{3}}{2},$$

  так как модуль и синус – непрерывные функции.
- Но из этого следует, что и

$$\lim_{n\to\infty} a_n = \lim_{n\to\infty}|\sin(f(n))| = \lim_{x\to+\infty}|\sin(f(x))| = \frac{\sqrt{3}}{2}.$$

■

B. Случайная дружба

В группе первокурсников ШАДа n ребят и они видят друг друга впервые. А в парке аттракционов сегодня дают скидку, если приходит компания ровно из k друзей, среди которых каждый дружит с каждым. Скидку получают все первокурсники из этой компании.
Дружба между любой парой однокурсников независимо возникает с вероятностью p. Макс, один из первокурсников, хочет пойти на аттракционы со скидкой.
Каким в среднем количеством способов он может это сделать?
Ответ: $C_{n-1}^{k-1}p^{C_k^2}$

Решение

Индикаторная техника. Матожидание суммы индикаторов успешных групп друзей.

- Ясно, что ответом служит математическое ожидание числа групп, состоящих из k ребят, включая Макса, в каждой из которых все попарно дружат друг с другом:

$$\mathbb{E}\,(\text{числа подходящих групп}) = \mathbb{E}\left(\sum_{t\in T}\mathbb{1}(\text{в } t \text{ все попарно знакомы})\right) = \sum_{t\in T}\mathbb{P}\,(\text{в } t \text{ все попарно знакомы})\,,$$

  где T – множество всевозможных ребят мощности k, включающих Макса, а $\mathbb{1}(A)$ – индикатор события A.

- Всего таких различных групп $|T| = C_{n-1}^{k-1}$, так как Макс зафиксирован.

- Вероятность, что в одной из таких групп каждый знает каждого $p^{C_k^2}$

- $$\mathbb{E}(\text{числа подходящих групп}) = |T| p^{C_k^2} = C_{n-1}^{k-1} p^{C_k^2}$$

■

C. Форма оператора

Рассмотрим некоторую билинейную функцию $Q : V \times W \to \mathbb{R}$ на вещественных линейных пространствах V и W. В базисах α и β Q имеет матрицу $Q_{\alpha,\,\beta}$:

$$Q(v, w) = v_\alpha^T \left(Q_{\alpha,\,\beta}\right) w_\beta,$$

где $v_\alpha,\ w_\beta$ – координаты векторов $v,\ w$ в соответствующих базисах.
Рассмотрим $Q_{\alpha,\,\beta}$ как матрицу некоторого линейного оператора в некотором базисе. Оказалось, что в некотором другом базисе его матрица B обладает следующим свойством: $B^2 = B$.
Найдите максимальную и минимальную возможную размерности собственного подпространства этого оператора, если известно, что Q в некоторой другой паре базисов $\omega,\ \gamma$

$$Q_{\omega,\,\gamma} = \begin{pmatrix} 2 & 0 & 2 & 3 \\ 0 & 0 & 0 & 2 \\ 4 & 0 & 4 & 5 \\ 2 & 0 & 2 & 2 \end{pmatrix}.$$

Ответ: 2

Решение

- Замена базиса у матрицы билинейной функции сохраняет лишь ранг.

$$\operatorname{rk}(Q_{\omega,\,\gamma}) = 2$$

.

- Оператор B является оператором проектирования ранга 2, а значит существует базис из собственных векторов, в котором он имеет диагональный вид с двумя 0 и двумя 1 на диагонали.

- А это, в свою очередь, означает, что все его собственные подпространства имеют размерность 2. ■

D. Фантом

На линейном пространстве размерности 4 над полем $\mathbb{R}$ задан класс линейных операторов C.
Класс C состоит из всех операторов обладающих следующим свойством:
Для любого оператора A из класса C выполнено:
- Все собственные значения A действительные.
- В одном из базисов исходного пространства φ: $\operatorname{rank}(A_\varphi) = 2$ и $\operatorname{tr}(A_\varphi A_\varphi^T) = 5$, где A_φ– матрица A в базисе φ.

Базис φ для каждого оператора из C свой.
Найдите $\sup_{A\in C} |\lambda(A)|$ и $\inf_{A\in C} |\lambda(A)|$, где $\lambda(A)$– максимальное по модулю собственное значение оператора A.
Ответ: $\inf_{A\in C} |\lambda(A)| = 0,\ \sup_{A\in C} |\lambda(A)| = \sqrt{5}$

Решение

- Матрица с действительными собственными значениями может быть приведена ортогональным преобразованием к верхнетреугольному виду.

- Норма Фробениуса при ортогональном преобразовании не изменится, а значит, мы имеем верхнетреугольную матрицу, сумма квадратов элементов которой равна 5.

- Возьмём верхнетреугольную матрицу:
$$\mathcal{A} = \begin{pmatrix} \sqrt{5-\varepsilon} & 0 \\ 0 & \sqrt{\varepsilon} \end{pmatrix}$$
Она удовлетворяет всем заданным свойствам.

- Устремим ε к 0 справа. Так как собственные числа у верхнетреугольных матриц стоят на диагонали, мы тем самым будем увеличивать максимальное по модулю собственное значение. Поэтому

$$\sup_{A \in C} |\lambda(A)| = \sqrt{5}$$

- Найдём похожим образом нижнюю грань:

$$\mathcal{A} = \begin{pmatrix} \sqrt{\varepsilon} & \sqrt{5 - 2\varepsilon} \\ 0 & \sqrt{\varepsilon} \end{pmatrix}$$

- Устремим ε к нулю справа. Получим, что:

$$\inf_{A \in C} |\lambda(A)| = 0.$$

- На самом деле можно построить матрицы порядка большего 2 и убедиться, что нижняя грань достижима. ■

E. Перебор

Известно, что случайный вектор $(X,\ Y)$ распределён нормально:

$$X, Y \sim N\left(\begin{pmatrix} 1 \\ 2 \end{pmatrix}, \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix}\right).$$

Найдите измеримую функцию $g(x)$, минимизирующую математическое ожидание

$$\mathrm{E}(-24Y^4 + (g(X))^2 + 2g(X)Y^2 + 4).$$

Решение

Напомним замечательный факт про многомерное нормальное распределение.
Пусть задан многомерный нормальный вектор

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \text{ размером } \begin{bmatrix} q \times 1 \\ (N - q) \times 1 \end{bmatrix}$$

$\boldsymbol{\mu}$ и $\boldsymbol{\Sigma}$ имеют вид

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix} \text{ размером } \begin{bmatrix} q \times 1 \\ (N - q) \times 1 \end{bmatrix}$$
$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \text{ размером } \begin{bmatrix} q \times q & q \times (N - q) \\ (N - q) \times q & (N - q) \times (N - q) \end{bmatrix}$$

тогда условное распределение $(\mathbf{x}_1 \mid \mathbf{x}_2) \sim N(\overline{\boldsymbol{\mu}}, \overline{\boldsymbol{\Sigma}})$ где

$$\overline{\boldsymbol{\mu}} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathbf{x}_2 - \boldsymbol{\mu}_2)$$

и ковариационной матрицы

$$\bar{\Sigma} = \Sigma_{11} - \Sigma_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

- Перепишем искомое выражение:

$$\mathrm{E}(-24Y^4 + (g(X))^2 + 2g(X)Y^2 + 4) = \mathrm{E}((Y^2 + g(X))^2 - 25Y^2 + 4)$$

- Поиск оптимальной g для данной функции эквивалентен поиску оптимальной g для

$$\mathrm{E}((Y^2 + g(X))^2)$$

- Это классическая задача и её решением является

$$g(x) = -\mathbb{E}\left(Y^2 \mid X = x\right)$$

- Найдём условное распределение $Y \mid X$. Вспомним, что есть формулы выше и запишем: $Y \mid X \sim N(X + 1, 1)$
- $g(x) = -1 - (x + 1)^2$
- $Y \sim N(2, 2)$
- $\mathbb{E}(g(Y)) = -1 - \mathbb{E}\left((Y + 1)^2\right) = 1 + 2 + 9 = -12$ ■

F. Маленькая компания

В компании 100 человек. Известно, что среди любых 10 есть трое попарно знакомых (кроме этих знакомств могут быть и еще какие-то). При каком наименьшем n гарантировано будет верно следующее утверждение: найдётся n человек таких, что любой из оставшихся знаком с кем-то из этих n?

Решение

Построим контрпример для 7 человек. Возьмем 7 изолированных вершин и еще полный граф на 93 вершинах. Тогда условие задачи выполнено (среди любых 10 вершин найдется треугольник), но нет никаких 7 вершин, к которым было бы ребро у любой из оставшихся. В самом деле, 7 изолированных вершин не подходят (к ним ни у кого нет ребер). А если взять рассмотреть любые 7 вершин, среди которых не все изолированные, то у оставшихся изолированных не будет ребер к этим семи.

Покажем, почему множество не более чем из 8 найдется. Рассмотрим максимальное по включению независимое множество, то есть такое множество вершин, что между ними нет ребер, но при добавлении к ним любой другой вершины это свойство нарушается. Пусть в нашем множестве k вершин. Заметим, что $k \leq 8$. В самом деле, если бы было $k \geq 10$, то это уже противоречие с условием (на любых 10 вершинах есть треугольник), а если $k = 9$, то добавим к нашему множеству любую из оставшихся вершин и получим противоречие с тем же условием. В то же время из максимальности нашего множества следует, что любая из оставшихся вершин имеет ребро с хотя бы одной вершиной нашего множества. ■

G. Обратный торрент

Рассмотрим систему из n серверов, расположенных в d дата-центрах. Серверы пронумерованы натуральными числами от 1 до n. На первом сервере расположен уникальный файл размера m GB. Наша задача — продублировать его на остальные серверы.

Для каждого сервера s известны входящая скорость передачи данных, исходящая скорость передачи данных и номер дата-центра, где расположен сервер. В этой задаче скорость передачи данных будем измерять в секундах, затрачиваемых на 1 GB получаемых или отправляемых данных. **В задаче используется нестандартная характеристика скорости.**

С сервера S на сервер T файл передается за время $m \cdot dc_{S_d T_d} \cdot \max(S_{out},\ T_{in})$, где dc_{ij} — некоторый коэффициент, задающий замедление передачи данных от дата-центра с номером i в дата-центр с номером j, S_{out} — минимальное время передачи 1 GB данных с сервера S, T_{in} — минимальное время получения 1 GB данные сервером T.

Информация передаётся по серверам в соответствии с порядком их номеров. Пока два сервера обмениваются данными, они оба заблокированы. В начальный момент времени 0 файл доступен только на первом сервере.

Будем считать, что обработка очереди копирования файлов происходит по следующему принципу:

- Сначала завершается операция копирования, которая должна завершиться в этот момент.
- Затем в порядке возрастания номеров серверов, получивших в данный момент времени файл, выводятся сообщения в лог.
- Затем формируется пул из серверов, с которых в этот момент может начаться копирование. Это все сервера, которые завершили передачу данных или получение.
- В порядке возрастания номеров доступных серверов они получают новое задание (в порядке возрастания номеров серверов для получения данных).
- Если в какой-то момент все сервера уже получили файл или получают его, то новых запусков копирования не происходит.

Выведите лог процесса копирования. Каждая строка должна содержать запись `Time X: server A received file from server B`. Строки лога должны быть отсортированы по возрастанию времени X, а при равенстве X по возрастанию серверов-получателей A.

Формат ввода:

В первой строке записано три целых числа n, d и m ($2 \leq n \leq 1\,000\,000$, $1 \leq d \leq 10$, $1 \leq m \leq 1000$).

В каждой из следующих n строк записаны параметры очередного сервера: три целых числа S_{in}, S_{out} и S_d ($1 \leq S_{in} \leq 1\,000\,000$, $1 \leq S_{out} \leq 1\,000\,000$, $1 \leq S_d \leq d$).

В следующих d строках записана целочисленная матрица dc_{ij} ($1 \leq dc_{ij} \leq 3$). Величина dc_{ij}, расположенная на j-й позиции i-й строки, задает коэффициент замедления передачи данных от дата-центра с номером i в дата-центр с номером j.

Формат вывода:

Выведите лог копирования данных.

Решение

- Будем использовать структуру данных, которая позволяет делать следующие операции: добавлять элемент в структуру, получать элемент из структуры с минимальным значением, удалять элемент из структуры с минимальным значением (считаем, что на множестве элементов задан полный порядок) и сравнивать два элемента можно за $O(1)$). Например, такой структурой данных является двоичная куча. Все эти операции она позволяет выполнять за $O(logn)$, где n – количество элементов в структуре в данный момент. Затраты памяти — $O(n)$.

- Будем хранить в куче события вида: в момент времени t, сервер x закончил пересылать файл серверу y, сравнивая события сначала по времени t, а при равенстве по номеру принимающего сервера y. Изначально добавим одно событие – сервер 1 отсылает файл серверу 2(время окончания пересылки здесь и далее равно времени начала + времени пересылки, правила вычисления которого подробно описаны в условии и которое можно по ним вычислить за $O(1)$ зная номера серверов).

- Теперь пока куча не пуста мы эмулируем описанный в условии алгоритм: берем событие, завершающееся следующим, и все остальные, которые завершатся в тот же момент, удаляем их все из кучи, выводим новые строчки лога, добавляем в «пул» все сервера, упомянутые в этих событиях и, пока не опустел список серверов в «пуле» или список серверов, которые еще не получили файл, запускаем трансфер файла и добавляем событие в кучу. Отдельно заметим, что для поддержания «пула» серверов можно так же использовать кучу. Для поддержания же списка серверов, еще не получивших файл, достаточно завести одну переменную, которая будет указывать на сервер с минимальным номером, еще не получившим файл.

- Сложность решения - $O(n \log n + d^2)$. Каждый сервер создаст одно событие, когда будет принимать файл, и каждое событие добавится и удалится из кучи один раз. Также линейное от n количество действий будет произведено с кучей для «пула» серверов. Все остальные действия занимает линейное от n время плюс надо считать матрицу $d \times d$.
  Если взять n за степень двойки, во все S_{in} положить 1, а S_{out} сделать равными произвольным числам, умноженным на большую константу для первых $\frac{n}{2}$ серверов, то сначала все первые $\frac{n}{2}$ серверов получат файл и будут готовы отсылать его остальным $\frac{n}{2}$ серверам(одновременно) и таким образом порядок получения файлов последними серверами будет соответствовать порядку чисел в S_{out}, то есть задача будет по факту сортировать эти числа. Поэтому ускорить его нельзя (в предположении произвольных значений S_{out}. Затраты памяти - $O(n + d^2)$ то есть равны размеру входа.

■

H. Игра

Пусть задан граф на n вершинах без единого ребра, его вершины занумерованы числами от 1 до n. Пусть также задан набор целых чисел $\{v_i\}_{i=1}^{n}$ со следующими свойством $\forall i\ 0 \le v_i < i$.
Стас идёт по вершинам графа в порядке возрастания номеров и смотрит на соответствующее текущей вершине под номером k число v_k. Он соединяет ребрами текущую вершину номером k с v_k предыдущими вершинами: $k - v_k, \cdots, k-1$. Найдите степени всех вершин графа и выведите их в порядке возрастания номера вершины. Степенью вершины графа называется число инцидентных этой вершине ребер.
Формат ввода:
В первой строке записано целое число n ($2 \le n \le 1\,000\,000$).
Во второй сроке записаны n целых чисел $v_1, v_2, \ldots, v_n$ ($0 \le v_i < i$).
Формат вывода:
Выведите n целых чисел $d_1, d_2, \ldots, d_n$, где d_i — степень вершины i.

Решение

- Степень каждой вершины равна количеству ребер, идущих из нее в вершины с меньшим номером, плюс числу ребер, идущих в вершины с большим номером. Из условия можно понять, что первая из этих величин нам и дана во входных данных.

- Итак, пусть эти величины будут равны $a_1, a_2, \ldots$ Когда Стас посещает вершину i, он увеличивает эти величины на 1 для вершин $k - v_i, \ldots, k-1$. Рассмотрим тогда величины $b_i = a_i - a_{i-1}$, где a_i для индексов i не попадающих в $[1; n]$ положим равными 0. Все что делает Стас на i-м шаге, это увеличивает b_{i-v_i} на 1 и уменьшает b_i на 1. С другой стороны, зная b легко восстановить $a : a_0 = b_0, a_i = a_{i+1} + b_i$ для $i > 0$.

- Поэтому решение состоит просто в том, чтобы сначала построить массив b, по правилам описанным выше, а потом по нему восстановить массив a. Ответ для каждой вершины – это сумма двух величин.

- Сложность решения по времени и памяти $O(n)$ что равно размеру входа и поэтому улучшить его нельзя. ■

0.1 Параллельно

На плоскости задана параметрическая кривая

$$\begin{cases} x(t) = t \\ y(t) = t + t\cos(\pi t) - 4 \end{cases},$$

где $t \in [1,\ 1.7]$.

Найдите точки на кривой, максимально удалённые от прямой, содержащей точки A и B с координатами: $\begin{pmatrix} 4 \\ 0 \end{pmatrix}$ и $\begin{pmatrix} 8 \\ 4 \end{pmatrix}$
В ответе введите сумму произведений координат полученных точек.

Решение

Сводится к решению уравнения

$$ctg(\pi t) = \pi t$$

на отрезке $[1, 1.7]$

Решается численно. ∎