

Максимальное количество баллов за олимпиаду — 600

Задание 1. Математика в чат-боте

Дима выбрал два натуральных числа a и b . Затем он отправил модели ИИ запрос — вычислить a^b . Он записал это выражение на бумажке, сфотографировал и загрузил фотографию. Из-за неаккуратного почерка модель распознала выражение как $a \cdot b$ и посчитала именно его. В итоге её ответ оказался меньше правильного на 110.

Какой ответ выдала модель?

Задание 2. Максимальный след по всем перестановкам

Матрицей $n \times m$ будем называть таблицу из чисел, состоящую из n строк и m столбцов. Умножение матриц выполняют по правилу «строка на столбец». Если

$$M = \begin{pmatrix} p & q \\ r & s \end{pmatrix}, \quad N = \begin{pmatrix} u & v \\ w & x \end{pmatrix},$$

то

$$MN = \begin{pmatrix} pu + qw & pv + qx \\ ru + sw & rv + sx \end{pmatrix}.$$

Суммой диагональных элементов (следом) матрицы называют число $tr \begin{pmatrix} p & q \\ r & s \end{pmatrix} = p + s$.

Дана матрица

$$A = \begin{pmatrix} -1 & 4 \\ 5 & 10 \end{pmatrix}.$$

Также даны числа $-4, -5, 20, 25$. Рассматриваются все 24 матрицы B вида

$$B = \begin{pmatrix} x & y \\ z & w \end{pmatrix},$$

в которых x, y, z, w — некоторая перестановка чисел $-4, -5, 20, 25$. Для каждой такой B рассмотрите произведения AB и BA .

- а) Найдите наибольшее возможное значение $tr(AB)$.
- б) Найдите наибольшее возможное значение $tr(BA)$.

Задание 3. Минимизация L_1 и L_2

Вы настраиваете простейшую регрессионную модель, которая всегда предсказывает одно и то же число c (константная модель). Дан набор истинных значений:

$$y = \{1, 2, 3, 9, 10, 10\}.$$

Рассмотрим две функции качества: $L_1(c) = \sum_i |y_i - c|$ и $L_2(c) = \sum_i (y_i - c)^2$.

- а) Найдите значение c_1 , минимизирующее $L_1(c)$. Если оптимальных значений несколько, в ответ запишите наименьшее.
- б) Найдите значение c_2 , минимизирующее $L_2(c)$. Если оптимальных значений несколько, в ответ запишите наименьшее.

Задание 4. Group By

Система электронного тестирования фиксирует результаты проверочных работ школьников, которые вы можете скачать в форматах XLSX, ODS или CSV. Файл содержит пять столбцов:

- **student_id** — идентификатор ученика (целое число);
- **subject** — предмет, по которому выполнен тест (строка);
- **score** — полученный балл (вещественное число);
- **cheat_flag** — подозрение на списывание (True/False);
- **attempt_no** — номер попытки (целое число, 1 означает первую попытку).

Выполните следующие действия с данными:

1. Очистите столбец **score**: пустые значения замените на 0, значения меньше 0 также замените на 0, значения больше 100 замените на 100.
2. Удалите строки, где **cheat_flag** = True.
3. Оставьте только строки, соответствующие первой попытке, то есть те, где **attempt_no** = 1.
4. Для каждого ученика вычислите его средний балл по предметам — это среднее всех значений **score**, которые остались после действий выше.

Сколько учеников имеют средний балл в диапазоне $60 \leq \text{avg_score} < 80$?

Задание 5. Дерево непринятия решений

Ограничение по времени: 1 секунда

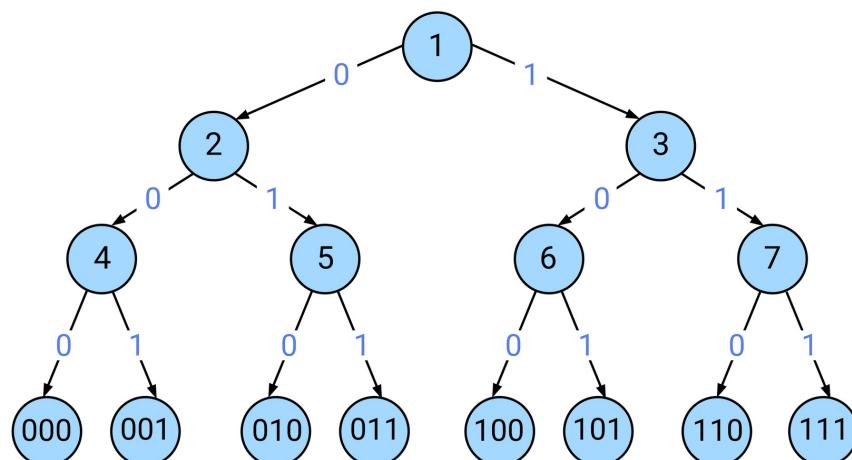
Ограничение по памяти: 256 мегабайт

В машинном обучении часто используют деревья решений. Каждая внутренняя вершина такого дерева соответствует некоторому вопросу, а каждое ребро — выбору ответа (да/нет). Таким образом, за несколько вопросов исходные данные могут быть разбиты на достаточно большое количество классов. Очередным проектом для Димы стало дерево непринятия решений. Его структура похожа на структуру решающего дерева. Это полное бинарное дерево глубины n . В каждой вершине, кроме вершин последнего уровня, хранится число p ($0 \leq p \leq 100$). Это число обозначает вероятность выбора: с вероятностью p процентов алгоритм выберет пойти влево и, соответственно, с вероятностью $100 - p$ процентов — вправо.

Договоримся: движение влево обозначим цифрой 0, а движение вправо — цифрой 1. Таким образом, каждая вершина нижнего уровня соответствует двоичной строке длины n (последовательности решений от корня до листа).

Вероятность получения этой строки равна произведению вероятностей всех выборов, сделанных на пути от корня до листа.

Дима уже написал структуру для такого дерева и хочет протестировать её. Для этого он создал дерево глубины 3. Значит, всего в нём 7 внутренних вершин. Каждая вершина имеет своё число p . Ниже показана схема расположения этих вершин:



Найдите вероятности всех двоичных строк длины 3 и выведите их в порядке неубывания вероятности. Если вероятности совпадают, строки должны выводиться в лексикографическом порядке.

Формат входных данных

В первой строке заданы 7 целых чисел p ($0 \leq p \leq 100$) — вероятности для вершин, как показано на рисунке.

Формат выходных данных

Выведите 8 строк. Каждая строка должна содержать двоичную строку длины 3. Строки должны идти в порядке неубывания вероятности. При равенстве вероятностей строки сравниваются лексикографически.

Примеры

стандартный ввод	стандартный вывод
40	011
90	110
20	001
90	101
100	010
70	100
0	000
	111

Замечание

В первом тестовом примере двоичные строки имеют следующие вероятности:

- 011 — 0.0
- 110 — 0.0
- 001 — 0.036
- 101 — 0.036
- 010 — 0.04
- 100 — 0.084
- 000 — 0.324
- 111 — 0.48

Критерий оценивания: точное совпадение ответа — 100 баллов

Максимальный балл за задание — 100

Задание 6. Тепловая карта

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Слава готовит постер на конференцию. К сожалению, сейчас его тепловая карта не влезает на постер: она слишком большая. Поэтому Слава решил выделить на текущей карте некоторый прямоугольный фрагмент и использовать его для презентации. Славина тепловая карта выглядит как таблица из n строк и m столбцов. Клетка на пересечении i -й строки и j -го столбца имеет цвет c_{ij} . Используются k цветов, которые пронумерованы от 1 до k . Пример аналогичной карты приведён справа. Слава хочет продемонстрировать весь размах значений, поэтому на выбранном фрагменте должна быть хотя бы одна клетка каждого цвета. При этом юный докладчик хочет минимизировать площадь карты, ведь ему нужно уместить её на постер.

Помогите ему: найдите прямоугольник, который можно будет вырезать из его карты так, чтобы на нём были клетки всех k цветов. Гарантируется, что на исходной тепловой карте присутствуют клетки всех k цветов.



Формат входных данных

В первой строке вводятся три натуральных числа n, m, k ($1 \leq n, m \leq 250, 1 \leq k \leq 20$).

В следующих n строках задаётся по m натуральных чисел c_{ij} ($1 \leq c_{ij} \leq k$).

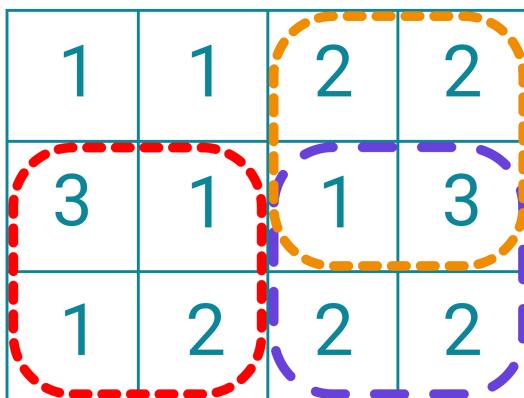
Формат выходных данных

Выведите 4 числа x_1, y_1, x_2, y_2 , задающие прямоугольник, который нужно вырезать. Прямоугольник задаётся своими верхней и нижней строками (x_1, x_2) и левым и правым столбцами (y_1, y_2). Если есть несколько способов вырезать график наименьшей площади, выведите любой.

Примеры

стандартный ввод	стандартный вывод
3 4 3	1 3 2 4
1 1 2 2	
3 1 1 3	
1 2 2 2	

Замечание



Все возможные варианты вырезать график наименьшей площади для первого примера.

Критерий оценивания: точное совпадение ответа — 100 баллов

Максимальный балл за задание — 100