

Чтобы начать решать задачи, зайдите в систему по адресу <https://pcms.itmo.ru> и нажмите кнопку **«Начать соревнование»**.

Около 200 человек по итогам районного этапа будут приглашены на региональный этап, который состоится 17 и 19 января 2026 года.

Обратите внимание, что в некоторых задачах необходимо отправить на проверку только файл с ответом. Вы можете получить этот ответ любым способом, например, посчитать вручную на листочке или написать программу. Файл должен иметь расширение **«.txt»** и может иметь любое имя, например, **«output.txt»**. Задачи с текстовым ответом можно отправлять на проверку не более 10 раз каждую.

Задача А. Род слов

Во многих европейских языках слова имеют род: мужской или женский (иногда также встречаются слова среднего рода или слова совсем без рода, в этой задаче такие слова не рассматриваются). Часто род слова можно определить по его окончанию (хотя исключения тоже встречаются очень часто). В этой задаче рассматривается синтетический язык, в котором род слова зависит от последних букв.

Будем рассматривать слова из строчных букв английского алфавита. Гласными считаются буквы **«a»**, **«e»**, **«i»**, **«o»**, **«u»**. Будем считать, что слово имеет женский род, если оно заканчивается на **«a»** (класс 1), либо на букву **«d»** (класс 2a), либо **«z»** (класс 2b), в этих двух случаях предпоследняя буква должна быть гласной, либо на буквосочетание **«ion»** (класс 3). В противном случае слово имеет мужской род.

Формат входных данных

На вход подана одна строка, содержащая слово, содержащее от 2 до 40 букв.

Формат выходных данных

Выведите **«f»**, если слово имеет женский род, либо **«m»**, если оно имеет мужской род.

Система оценки

В этой задаче 20 тестов, каждый оценивается в 5 баллов.

Примеры

стандартный ввод	стандартный вывод
gato	m
rata	f
nation	f

Задача В. Род слов — обучаем ИИ

Рассмотрим синтетический язык из предыдущей задачи.

Компания, разрабатывающая программу для работы с этим языком, хочет обучить нейросеть распознавать род слов. Вам необходимо создать синтетические данные для обучения этой нейросети.

В этой задаче вам необходимо предоставить на проверку файл с расширением **«.txt»**. Вы можете самостоятельно вручную создать этот файл или написать программу, которая его создаст.

Формат выходных данных

Файл для обучения должен содержать размеченные данные. Выведите 120 строк, из которых 60 задают слова мужского рода, а 60 — женского. В файле должно быть 20 слов женского рода класса 1, по 10 классов 2a и 2b, и 20 слов класса 3. Среди слов классов 2a и 2b должно быть поровну слов для каждой гласной перед **«d»** и **«z»**, соответственно.

Каждая строка должна содержать слово, затем пробел и букву **«m»**, если слово мужского рода, либо **«f»**, если слово женского рода.

Все слова в файле должны быть различны.

Система оценки

Если все условия выполнены, вы получаете 100 баллов.

Если выходной файл не соответствует формату, хотя бы одно слово размечено неверно (указан неверный род **«m/f»**), или он содержит повторяющиеся слова, вы получаете 0 баллов. Рекомендуем сдать задачу А перед решением этой задачи, чтобы научиться корректно определять род слов.

Если выходной файл содержит 60 слов мужского рода и 60 слов женского рода, причем хотя бы по одному слову женского рода каждого класса, вы получаете 70 баллов.

Если выходной файл содержит хотя бы 10 слов каждого рода, и для каждого класса женского рода хотя бы одно слово, вы получаете 40 баллов.

Если выходной файл содержит хотя бы 10 слов каждого рода, вы получаете 20 баллов.

Иначе вы получаете 0 баллов.

В примере приведен пример правильно отформатированного файла, который, однако, содержит только три слова, поэтому оценивается в 0 баллов.

Пример

output.txt
gato m rata f nation f

Задача С. Обновление базы данных

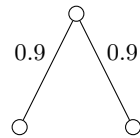
Для задач машинного обучения используется кластер из серверов. Кластер организован в виде бинарного дерева: центральный сервер является корнем дерева и имеет уровень 0. Каждый сервер уровня i для i от 0 до $k-1$ соединен с двумя дочерними серверами уровня $i+1$. Серверы на уровне k — рабочие, у них нет дочерних серверов.

Обновление базы данных поступает на центральный сервер и далее распространяется по остальным серверам. Каждый сервер, получив обновление, передает его далее своим дочерним серверам.

Из-за высокой нагрузки передача информации между серверами не всегда происходит успешно. Известно, что вероятность того, что обновление будет успешно передано от сервера к его дочернему серверу, составляет 0.9.

Необходимо посчитать среднее число рабочих серверов, которые получают обновление.

Например, если $k = 1$, то есть два рабочих сервера, непосредственно являющиеся дочерними для центрального.



- с вероятностью 0.9^2 оба сервера получают обновление;
- с вероятностью $0.9 \cdot 0.1$ первый рабочий сервер получит обновление, а второй — нет;
- с вероятностью $0.1 \cdot 0.9$ первый рабочий сервер не получит обновление, а второй — получит;
- с вероятностью 0.1^2 оба сервера не получают обновление.

Среднее число серверов, которые получают обновление, равно

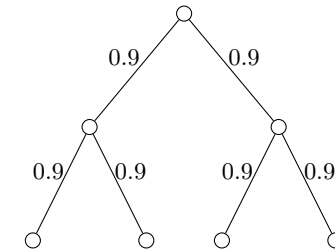
$$2 \cdot 0.9^2 + 1 \cdot 0.9 \cdot 0.1 + 1 \cdot 0.1 \cdot 0.9 + 0 \cdot 0.1^2 = 1.62 + 0.09 + 0.09 + 0 = 1.8.$$

Формат выходных данных

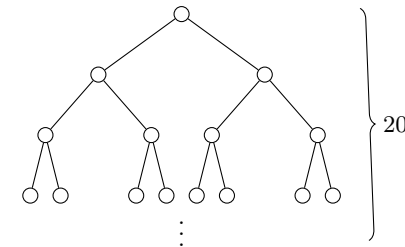
В этой задаче вам необходимо предоставить на проверку файл с расширением «.txt». Вы можете самостоятельно вручную создать этот файл или написать программу, которая его создаст.

В файле должны быть одна или две строки.

Первая строка должна содержать среднее количество рабочих серверов, которые получают обновление, если $k = 2$. Если это значение верно, вы получите 50 баллов.



Вторая строка может содержать среднее количество рабочих серверов, которые получают обновление, если $k = 20$. Если это значение верно, вы получите еще 50 баллов.



Ваш ответ будет засчитан, если он имеет абсолютную или относительную погрешность не больше 10^{-6} .

Пример выходного файла ниже содержит корректное форматирование, но некорректные значения.

Пример

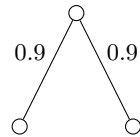
output.txt
1.8
180.0

Задача D. Обновление базы данных - часть 2

Рассмотрим кластер из серверов из предыдущей задачи.

В этой задаче условие такое же, но необходимо посчитать вероятность, что число рабочих серверов, которые получают обновление, составляет хотя бы половину всех рабочих серверов.

Например, если $k = 1$, то есть два рабочих сервера, непосредственно являющиеся дочерними для центрального.



- с вероятностью 0.9^2 оба сервера получают обновление;
- с вероятностью $0.9 \cdot 0.1$ первый рабочий сервер получит обновление, а второй — нет;
- с вероятностью $0.1 \cdot 0.9$ первый рабочий сервер не получит обновление, а второй — получит;
- с вероятностью 0.1^2 оба сервера не получают обновление.

Таким образом, хотя бы половина, то есть хотя бы один сервер, получит обновление с вероятностью

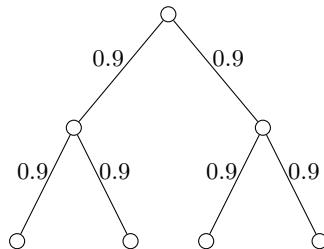
$$0.9^2 + 0.9 \cdot 0.1 + 0.1 \cdot 0.9 = 0.81 + 0.09 + 0.09 = 0.99$$

Формат выходных данных

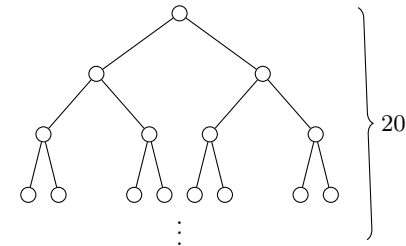
В этой задаче вам необходимо предоставить на проверку файл с расширением «.txt». Вы можете самостоятельно вручную создать этот файл или написать программу, которая его создаст.

В файле должны быть одна или две строки.

Первая строка должна содержать вероятность, что хотя бы половина рабочих серверов получит обновление, если $k = 2$. Если это значение верно, вы получите 50 баллов.



Вторая строка может содержать вероятность, что хотя бы половина рабочих серверов получат обновление, если $k = 20$. Если это значение верно, вы получите 50 баллов.



Ваш ответ будет засчитан, если он имеет абсолютную или относительную погрешность не больше 10^{-6} .

Пример выходного файла ниже содержит корректное форматирование, но некорректные значения.

Пример

output.txt
0.99
0.9

Задача Е. 1-НОТ кодирование

1-НОТ кодирование — способ кодирования объектов векторами для задач машинного обучения, когда между разными объектами нет какой-либо численной связи. Пусть у нас есть n объектов, тогда можно закодировать объекты n -мерными векторами следующим образом: i -й объект кодируется вектором, у которого i -я координата равна 1, а все остальные равны 0.

Задан набор объектов и несколько запросов. Для каждого запроса выведите 1-НОТ код соответствующего запросу объекта.

Формат входных данных

Первая строка ввода содержит число n — количество объектов ($1 \leq n \leq 100$). Последующие n строк содержат по одному слову, состоящему из строчных букв английского алфавита, — наименования объектов. Каждое наименование объекта представляет собой строку не длиннее 100 символов. Все наименования объектов различны.

Следующая строка содержит целое число q — количество запросов ($1 \leq q \leq 100$). Следующие q строк содержат по одному слову: объекты в запросах. Гарантируется, что все объекты в запросах соответствуют одному из n описанных объектов.

Формат выходных данных

Для каждого запроса выведите n координат соответствующего вектора, разделенные пробелами.

Пример

стандартный ввод	стандартный вывод
4	1 0 0 0
cat	0 0 0 1
dog	0 0 0 1
mouse	0 0 1 0
bird	0 1 0 0
5	
cat	
bird	
bird	
mouse	
dog	

Задача F. Квадратный корень

Напомним, как работает умножение матриц. Пусть заданы две матрицы 2×2 :

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \text{ и } B = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}.$$

Их произведением называется матрица C , которая вычисляется по следующей формуле:

$$C = AB = \begin{pmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{pmatrix}.$$

Задана матрица

$$A = \begin{pmatrix} a & b \\ 0 & c \end{pmatrix},$$

где a , b и c — вещественные числа.

Требуется найти такую матрицу X , чтобы выполнялось равенство $X^2 = A$. Если подходящих матриц X несколько, можно вывести любую из них.

Формат входных данных

На вход подаются три вещественных числа: a , b , c ($1 \leq a, c \leq 1000$, $-1000 \leq b \leq 1000$).

Формат выходных данных

Если искомая матрица X существует, выведите четыре вещественных числа: x_{11} , x_{12} , x_{21} , x_{22} . Если подходящих матриц несколько, можно вывести любую из них.

При проверке, что ваша матрица подходит, проверяющая программа будет сравнивать вещественные числа с точностью 10^{-6} . Рекомендуется выводить не менее 12 значащих цифр.

В противном случае выведите четыре нуля.

Система оценки

В этой задаче 100 тестов, каждый оценивается независимо в 1 балл.

Пример

стандартный ввод	стандартный вывод
1.0 0.0 1.0	1.0 0.0 0 1.0

Задача G. Ограничение скорости

При разработке автономного транспорта важно следить за соблюдением скорости на дороге. В этой задаче изложены немного упрощенные правила, устанавливающие ограничения скорости в России. Условие этой задачи не следует использовать как замену правилам дорожного движения.

В России бывают три зоны ограничения скорости по умолчанию:

- населенный пункт — ограничение скорости 60 км/ч. Для обозначения начала населенного пункта используется табличка с именем этого населенного пункта на белом фоне;
- вне населенного пункта — ограничение скорости 90 км/ч. Для обозначения конца населенного пункта используется табличка с перечеркнутым именем этого населенного пункта на белом фоне;
- автомагистраль — ограничение скорости 110 км/ч. Для обозначения начала автомагистрали используется схематическое изображение магистрали на зеленом фоне. Для обозначения окончания автомагистрали используется аналогичный зачеркнутый знак. После окончания автомагистрали, если она начиналась в населенном пункте, продолжается населенный пункт; если она начиналась вне населенного пункта, дорога продолжается вне населенного пункта.

Для изменения ограничения скорости по умолчанию используются знаки временного ограничения скорости. Если на дороге установлен знак с числом x , то после этого знака максимальная скорость на дороге равна x км/ч. Обратите внимание, что x может быть как меньше ограничения скорости по умолчанию в текущей зоне (для того, чтобы замедлить проезжающие автомобили), так и больше (если дорога позволяет двигаться с более высокой скоростью).

Для отмены временного ограничения скорости используются знаки отмены ограничения скорости с зачеркнутым числом или отмены всех ограничений. Также временное ограничение скорости отменяется после любого перекрестка, начала или конца населенного пункта или автомагистрали. После отмены ограничения скорости максимальная скорость становится равна скорости по умолчанию для текущей зоны; старые ограничения не возвращаются.

Вы разрабатываете программу для управления автономным автомобилем. Система компьютерного зрения передает вам последовательность событий: знаков, которые встречаются на дороге, а также перекрестков. После каждого события необходимо вывести максимальную скорость, с которой можно двигаться после этого события. Считайте, что исходно автомобиль находится в населенном пункте и временного ограничения скорости нет.

Формат входных данных

Первая строка ввода содержит число n — количество событий ($1 \leq n \leq 1000$). Далее следует n строк, содержащих описание событий.

Каждое описание находится в одной строке и начинается со слова — типа события.

- «city» — начало населенного пункта;

- «nocity» — конец населенного пункта;
- «highway» — начало автомагистрали;
- «nohighway» — конец автомагистрали;
- «limit» — начало временного ограничения скорости, затем через пробел написано само ограничение, целое число от 5 до 130 включительно;
- «nolimit» — конец временного ограничения скорости или конец всех ограничений;
- «cross» — перекресток.

Гарантируется, что последовательность знаков корректна: начало населенного пункта встречается только, если автомобиль находится вне населенного пункта; конец населенного пункта встречается только, если автомобиль находится в населенном пункте; начало автомагистрали встречается только, если автомобиль не находится на магистрали; конец автомагистрали встречается только, если автомобиль находится на автомагистрали. Однако отмена всех ограничений может встретиться, и если ранее не было введено ограничений скорости (возможны другие ограничения).

Формат выходных данных

Следует вывести n строк: ограничение скорости после каждого события.

Пример

стандартный ввод	стандартный вывод
9	60
cross	90
nocity	30
limit 30	90
nolimit	110
highway	130
limit 130	90
limit 90	110
nolimit	90
nohighway	

Задача Н. Высокий спрос

Система распределения заказов такси учитывает высокий спрос и поднимает цену, чтобы удовлетворить заказы пассажиров, готовых заплатить больше, и привлечь на работу больше таксистов. Будем рассматривать поездки внутри района, где стоимость одинакова для всех заказов. Стоимость является целым положительным числом.

Будем считать, что такси вызывают n пассажиров, пронумерованных от 0 до $n - 1$, i -й пассажир готов заплатить за поездку не более $A_i + B$. На работу потенциально готовы выйти m таксистов, пронумерованных от 0 до $m - 1$, j -й таксист готов выполнить заказ, если получит за его выполнение хотя бы $Cj + D$. Каждый таксист в рамках этой задачи выполнит не более одного заказа.

Какую стоимость поездки x следует установить, чтобы максимальное число пассажиров смогли воспользоваться поездкой на такси?

Формат входных данных

Ввод содержит целые числа n, m, A, B, C, D , по одному на строке, каждое число лежит в диапазоне от 1 до 10^9 , включительно.

Гарантируется, что $A(n - 1) + B \geq D$.

Формат выходных данных

Выведите одно целое положительное число x : какую стоимость заказа следует установить, чтобы максимальное число пассажиров смогли воспользоваться такси. Если подходящих ответов несколько, следует вывести максимальный из них.

Система оценки

В этой задаче 25 тестов, каждый оценивается в 4 балла. В 10 из этих тестов все числа во вводе не превышают 100. Еще в 5 из этих тестов $n, m \leq 5\,000$.

Пример

стандартный ввод	стандартный вывод
3	100
3	
30	
70	
15	
80	

Замечание

В примере есть три пассажира, готовые заплатить за поездку, соответственно, 70, 100 и 130. Есть три таксиста, которые готовы выполнить заказ, если получают за него, соответственно, 80, 95 и 110. Если установить стоимость поездки между 95 и 100, то два пассажира смогут воспользоваться поездкой.

Задача I. Анализ оценок

В этой задаче вам необходимо проанализировать оценки учеников, заданные в виде выгрузки из базы данных в формате «.csv».

Вы хотите проверить гипотезу, что в младших классах девочки учатся лучше мальчиков, а в старших — наоборот. Необходимо выбрать всех учеников, у которых есть хотя бы три оценки. Остальных учеников в этой задаче следует полностью игнорировать. Для каждого из этих учеников необходимо найти среднюю оценку: среднее арифметическое их баллов. После этого для каждого класса i , где были выбраны как мальчики, так и девочки, необходимо вычислить среднее значение средней оценки среди девочек в этом классе F_i и среднее значение средней оценки среди мальчиков в этом классе M_i . После этого необходимо найти класс, в котором величина $F_i - M_i$ максимальна D_{max} (девочки учатся лучше мальчиков) и минимальна D_{min} (мальчики учатся лучше девочек).

В качестве ответа необходимо сдать текстовый файл с двумя вещественными числами: D_{max} и D_{min} .

Формат входных данных

В этой задаче два файла с данными. Их можно скачать на вкладке «Файлы» в тестирующей системе.

Файл «grades.csv» содержит оценки. Файл содержит несколько строк с данными, поля в каждой строке разделены символом «;». Формат строки файла:

ID оценки;ID ученика;ID предмета;Оценка

Все ID представляют собой положительные целые числа не больше 10^9 , оценка представляет собой целое число от 1 до 5. ID оценки и ID предмета вам не требуются, их можно игнорировать.

Файл «students.csv» содержит информацию об учениках. Файл содержит несколько строк с данными, поля в каждой строке разделены символом «;». Формат строки файла:

ID ученика;Фамилия ученика;Имя ученика;Пол ученика;Класс ученика

ID ученика тот же, что и в файле «grades.csv», фамилия и имя — строки из английских букв, они вам не требуются, их можно игнорировать. Пол ученика — буква «f» для девочек, «m» для мальчиков. Класс ученика — строка из букв и цифр.

Формат выходных данных

Необходимо отправить на проверку текстовый файл с расширением «.txt», он должен содержать два вещественных числа: D_{max} и D_{min} . Ответ будет засчитан, если он будет иметь абсолютную или относительную погрешность не больше 10^{-6} .

Пример выходного файла имеет корректный формат, но содержит некорректный ответ.

Система оценки

Если два выведенных числа верны, вы получите 100 баллов, иначе 0.

Пример

output.txt
0.239030 -0.566369