

Задача А. Автомобили в Байтландии

Имя входного файла: `cars.in`
Имя выходного файла: `cars.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Ни для кого не секрет, что в Байтландии выпускается очень много различных моделей автомобилей. Хотя со временем устаревшие модели снимают с производства, им на смену приходит все больше новых. Каждая модель автомобиля характеризуется различными параметрами, и один из самых важных — стоимость автомобиля.

Футболисты местного футбольного клуба «Меткий баг» — довольно странные люди, и при покупке автомобиля в первую очередь руководствуются именно стоимостью автомобиля. Они не станут покупать слишком дорогую или дешевую модель. Футболисты считают модель слишком дорогой, если ее стоимость больше средней арифметической стоимости всех моделей, выпускаемых на данный момент, и слишком дешевой, если ее стоимость меньше. Модели со стоимостью, строго равной этому среднему арифметическому, напротив, пользуются большим спросом у футболистов. Компанию, выпускающую автомобили в Байтландии, заинтересовал вопрос: сколько моделей будут пользоваться спросом среди Байтландских футболистов в определенный момент?

Формат входного файла

В первой строке входного файла задано количество запросов n ($1 \leq n \leq 10^5$). В следующих n строках заданы запросы. Запросы бывают трех видов:

- запрос на добавление модели в производство в виде «+ k », где k — стоимость новой модели ($0 \leq k \leq 10^{13}$)
- запрос на удаление наиболее устаревшей модели из производства в виде «-». Наиболее устаревшей моделью считается та, которая начала выпускаться раньше, чем все остальные, выпускаемые в данный момент. Гарантируется, что при поступлении этого запроса в производстве есть хотя бы одна модель.
- запрос на вывод количества моделей в производстве, стоимость которых равна среднему арифметическому стоимостей всех автомобилей, выпускаемых в данный момент. Запрос поступает в виде строки «?». Гарантируется, что в момент запроса в производстве есть хотя бы одна модель.

Изначально в производстве нет ни одной модели.

Формат выходного файла

На каждый запрос вида «?» выведите ответ в отдельной строке. Ответы выводите в порядке следования запросов во входном файле.

Примеры

<code>cars.in</code>	<code>cars.out</code>
10	1
+ 1	0
+ 4	1
+ 3	
+ 4	
?	
-	
-	
?	
-	
?	

Задача В. Закат

Имя входного файла:	fall.in
Имя выходного файла:	fall.out
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Долгие переговоры руководства ФК «Закат» увенчались успехом: Брюс Бэннер присоединился к команде. Как и положено звезде мирового футбола, он получал немаленькую зарплату.

Однако, одному из прежних игроков команды это не понравилось. Стало ясно, что не избежать волны требований увеличения заработка. После некоторого раздумья, великий тренер Чулано Спагетти нашел способ решить эту проблему.

Всего в команде «Закат» n игроков. Некоторые из них имеют российское гражданство, все остальные — легионеры. К требованиям легионеров тренер относится с осторожностью и предпочтет увеличить им зарплату, чтобы не упустить ценного игрока. А вот российских игроков тренер будет стимулировать снижать их запросы следующим действием: в ближайшем матче не выпускать на поле того игрока с российским гражданством, который потребует себе наибольшую зарплату. Теперь ваша задача определить, кто из команды не выйдет на поле, зная фамилии игроков, их гражданство и зарплату, которую они требуют.

Формат входного файла

В первой строке задано число футболистов n ($1 \leq n \leq 100$). Далее следуют n строк, содержащих описание футболистов. Каждое описание состоит из перечисленных через пробел зарплаты, требуемой этим игроком, его фамилии и гражданства. Фамилия и гражданство — строки, содержащие только строчные и прописные латинские символы. Длина фамилии и гражданства не превышает 20 символов. У игроков, имеющих российское гражданство, в соответствующем параметре значится «Russia». Зарплата, которую требует игрок — положительное число, не превышающее 10^9 . Гарантируется, что в команде есть хотя бы один игрок с российским гражданством.

Формат выходного файла

Выведите фамилию футболиста, который не выйдет на поле. Гарантируется, что такой игрок всегда один.

Примеры

fall.in	fall.out
5 5000000 Banner Brazil 1000000 Malafeev Russia 3000000 Denisov Russia 2000000 Kerzhakov Russia 4000000 Criscito Italy	Denisov

Задача С. Футбол

Имя входного файла: `football.in`
Имя выходного файла: `football.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Хаус с Уилсоном, смотря финальный матч Кубка MLS по футболу между командами LAG и DCU, в очередной раз поспорили. Хаус говорил, что победит команда LAG, Уилсон же говорил, что команда DCU. Когда закончился первый матч финальной серии (а финальная серия состоит из двух матчей), Уилсон задумался какие исходы второго матча помогут Уилсону выиграть спор.

Напомним, что команда выигрывает в серии из двух матчей, если по сумме двух игр она забила больше мячей, чем команда соперника. Если же обе команды забили суммарно одинаковое количество голов, то победителем считается команда, которая забила больше мячей ну чужом поле. Если же в гостях они также забили одинаковое количество голов, то проводится два дополнительных тайма. В рамках данной задачи будем считать этот исход ничейным.

Известно, что первый матч играют на поле LAG, а второй — на поле DCU. Кроме того, Уилсон предположил, что никакая команда не забьёт более девяти голов. Ваша задача — помочь Уилсону посчитать, сколько существует исходов, при которых кубок выиграет команда LAG и сколько существует исходов, при которых кубок выиграет команда DCU. При этом вам будет дан результат первого матча.

Формат входного файла

В первой строке дано целое число N ($1 \leq N \leq 100$) — количество тестов. В следующих N строках дан результат в первом матче между командами в формате `a:b`, где `a` — количество голов забитое командой LAG и `b` — количество голов забитое DCU.

Формат выходного файла

В каждой из N строк вывести по два числа: количество исходов, при которых выиграет команда LAG и количество исходов, при которых выиграет команда DCU.

Примеры

football.in	football.out
3	62 37
2:1	0 99
0:9	53 46
1:1	

Задача D. Назначения

Имя входного файла: `prescription.in`
Имя выходного файла: `prescription.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Как известно, доктор Хаус очень не любит ждать. Особенно он не любит, когда его пациентам необходимо ждать в очереди на операцию. Обычно с этим проблем не возникает, так как руководство больницы всегда готово пойти на встречу лучшему врачу. Но сейчас, когда Форман уехал на конференцию, Хаусу приходится идти на отчаянные меры.

Недавно в больнице установили новую систему регистрации оперируемых больных. Хаусу пришлось нанять хакера, который взломал эту систему. Выяснилось, что в базе каждое назначение на операцию хранится в виде строки, которая может содержать маленькие латинские буквы, цифры и символ подчеркивания. Однако хакер, в силу невысокой квалификации, может изменять назначение в базе, только удаляя из него все вхождения некоторого символа. Кроме того, оказалось, что из каждой строки в базе можно удалить вхождения только одного символа, так как иначе она признается недействительной.

Доктор Хаус выбрал запись, которую он хочет изменить, и теперь ему интересно, какая лексикографически минимальная строка может из нее получиться.

Формат входного файла

В первой строке входного файла находится описание назначения на операцию, которое хочет исправить Хаус — строка s ($1 \leq |s| \leq 10^6$), состоящая из маленьких латинских букв, цифр и символов подчеркивания.

Формат выходного файла

В выходной файл выведите лексикографически минимальную строку, которая может получиться у доктора Хауса.

Примеры

<code>prescription.in</code>	<code>prescription.out</code>
<code>house_g_0101_first_january_angioplasty</code>	<code>hose_g_0101_first_janary_angioplasty</code>
<code>khoukse_k_1012_tenth_december_endoscopy</code>	<code>house__1012_tenth_december_endoscopy</code>

Задача Е. Стадион

Имя входного файла: `stadium.in`
Имя выходного файла: `stadium.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В Берляндии скоро пройдет чемпионат мира по футболу, и сейчас активно ведется его подготовка. Для проведения особо важных матчей и церемоний открытия и закрытия планируется построить новый стадион, который должен стать самым большим из известных человечеству. Однако, если стадион будет очень большим, то зрители могут рассаживаться очень долго. Для решения этой проблемы был создан специальный отдел оптимизации стадиона.

По проекту, зрительские места на стадионе будут разбиты на несколько одинаковых секторов. Сектор представляет из себя несколько рядов кресел по m кресел в каждом ряду с двумя проходами по бокам, так что к каждому месту можно будет пройти справа или слева. В каждом ряду кресла пронумерованы слева направо. Способом рассадки зрителей в одном ряду отдел называет перестановку чисел от единицы до m , соответствующую порядку, в котором зрители занимают свои места в этом ряду. Хорошим способом назовем такой способ, при котором никакому зрителю не придется проходить на пути к своему месту мимо уже сидящего человека. Вас же попросили посчитать, сколько существует различных таких способов. Считается, что очередной зритель начинает идти к своему месту только тогда, когда предыдущий за ним (если такой был) уже занял свое место.

Формат входного файла

В входном файле содержится одно целое число m ($1 \leq m \leq 10^{18}$) — число мест в каждом ряду.

Формат выходного файла

В выходной файл выведите одно число — число различных хороших способов рассадки зрителей по модулю $10^9 + 7$.

<code>stadium.in</code>	<code>stadium.out</code>
1	1
3	4

Задача F. Вирусы

Имя входного файла: `viruses.in`
Имя выходного файла: `viruses.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Пока доктору Хаусу снится страшный сон об эпидемии, его бывший подчиненный Эрик Форман совершает прорыв в науке! Сейчас он занимается изучением воздействия вирусов на различный типы тканей.

Форман уже выяснил, что ткань может быть представлена как последовательность клеток, каждая из которых характеризуется своей резистентностью. Каждый вирус можно охарактеризовать одним натуральным числом K — его заразностью.

У вируса с заразностью K , попавшего в какую-то клетку, есть три последовательных стадии жизни: инкубация, распространение и существование. Вирус в ткани живет по следующим правилам:

- изначально он находится только в клетке с номером один в стадии распространения
- в стадии инкубации и существования вирус спокойно поглощает свою клетку и никак не влияет на все остальные
- стадия инкубации вируса в клетке номер i переходит в стадию распространения ровно в тот момент, когда стадия распространения в клетке номер $i - 1$ заканчивается и переходит в стадию существования
- в стадии распространения вирус нападает на K клеток с наименьшими номерами, в которых вируса еще нет. Если резистентность какой-то из этих k клеток больше, чем резистентность клетки, из которой вирус распространяется, то вся ткань вырабатывает иммунитет, и вирус ее покидает. В противном же случае вирус поселяется в этих клетках и в них начинается стадия инкубации
- в случае успешного заражения K клеток, стадия распространения заканчивается и начинается в следующей клетке

Понятно, что каждая ткань будет заражена далеко не всеми вирусами. Сейчас же Форман хочет ответить на вопрос: какова минимальная заразность вируса, который, попав в изучаемую ткань, сможет полностью ее захватить.

Формат входного файла

В первой строке дано целое число N ($1 < N \leq 5000$) — количество клеток в ткани. Во второй строке дано N целых чисел a_i ($1 \leq a_i \leq 10^9$), обозначающих резистентности клеток ткани.

Формат выходного файла

В первой строке выведите единственное число K — минимальную заразность вируса, способного поглотить исследуемую ткань полностью. Если же такого вируса не существует — выведите 0.

Примеры

<code>viruses.in</code>	<code>viruses.out</code>
5 1 2 3 4 5	0
5 5 4 2 3 1	2

Задача G. Легкое слово

Имя входного файла: `word.in`
Имя выходного файла: `word.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Доминика Петрова прилетела в США из Украины с целью получить гражданство и зажить лучшей жизнью. В Америке она вышла замуж за Грегори Хауса и стала Доминикой Хаус. Поначалу ей было очень сложно освоиться в штатах. Еще бы, ведь фиктивный муж не очень стремился ей помогать. Поэтому Доминике пришлось самой изучать сложный английский язык.

По мере того, как Доминика узнавала все новые и новые слова, она поняла, что некоторые слова ей учить проще. Поэтому она ввела понятие сложности слова. Сложность слова — число, равное сумме двух параметров: отклонение слова от первой буквы и от последней. Отклонение слова от буквы s считается так: для каждой буквы считается модуль разности его позиции в алфавите с позицией буквы s , после чего берется максимум из всех этих величин.

Также Доминика заметила, что после нескольких циклических сдвигов слово переходит само в себя. Теперь ей стало интересно, сколько циклических сдвигов с минимальной сложностью ей встретится до того, как сдвиг строки первый раз совпадет с исходной строкой.

Формат входного файла

Во входном файле дано одно слово, состоящее из строчных букв английского алфавита. Длина слова не превосходит 10^6 .

Формат выходного файла

В выходной файл выведите два числа: минимальную сложность циклического сдвига, который встретится Доминике до того, как строка перейдет в себя, и число таких сдвигов.

Примеры

<code>word.in</code>	<code>word.out</code>
abcabc	3 2