

Задача А. Смотровая площадка

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

А вы хотели бы посмотреть на Екатеринбург с высоты птичьего полёта?

Антей — это современный комплекс, сочетающий в себе деловой, развлекательный и торговый центры, находящийся в самом центре Екатеринбурга. Этот комплекс включает в себя два высотных здания: одно высотой 19 этажей, другое — 50 этажей. 50-этажное новое здание ещё не достроено и пока не используется, поэтому смотровая площадка находится на крыше старого, 19-этажного, здания.

К сожалению, со смотровой площадки вам не удастся увидеть весь Екатеринбург, так как рядом стоит огромное недостроенное новое здание Антея. С первого взгляда может показаться, что оно скрывает от вас половину города, но это не так. Посчитайте, какая часть города в действительности закрыта новым зданием Антея от посетителей смотровой площадки.

Город Екатеринбург имеет форму круга радиуса R с центром в точке $(0, 0)$. Можно считать, что весь город, за исключением возвышающихся зданий Антея, расположен в горизонтальной плоскости. Оба высотных здания Антея имеют цилиндрическую форму. Основание старого здания — круг радиуса r_{old} с центром в точке $(0, 0)$, основание нового здания — круг радиуса r_{new} с центром в точке (x, y) . Оба здания целиком находятся внутри Екатеринбурга и не имеют общих точек. Посетители могут свободно гулять по смотровой площадке и осматривать город из любой её точки. Точка города *не видна* наблюдателю, если отрезок, соединяющий её с положением наблюдателя на смотровой площадке, имеет хотя бы одну общую точку с новым зданием Антея. Точка города считается *закрытой* от наблюдателя, если она не видна ни из одной точки на смотровой площадке.

Формат входного файла

В первой строке через пробел записаны целые числа R, r_{old}, r_{new} ($1 \leq r_{old} < r_{new} < R \leq 1000$). Во второй строке через пробел записаны целые числа x и y .

Формат выходного файла

Выведите часть города, закрытую от наблюдателя 50-этажным зданием Антея, в процентах, с точностью не менее шести знаков после десятичной точки.

Пример

input.txt	output.txt
10 1 3 -1 4	20.9346176061

Задача В. Книга Сандро

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Прошло уже немало лет с тех пор, как Лич Сандро ушёл на заслуженный отдых. Иногда по вечерам, когда ему становится совсем тоскливо, он берёт в руки книгу, которую ему подарили воспитанники-маги по случаю выхода на пенсию.

Вот и сейчас великий маг взял с полки книгу и углубился в чтение. В одной из глав рассказывалось про знаменитое открытие Сандро — много лет назад он придумал универсальное заклинание. Оказалось, что любая его подстрока (последовательность подряд идущих букв) тоже является заклинанием, а сила любого заклинания равна количеству раз, которое это заклинание встречается в универсальном (например, строка «ue» встречается в строке «queue» дважды, а строка «aba» в строке «abababa» — трижды).

Сейчас у Сандро много свободного времени, и он решил найти самое сильное заклинание. Помогите ему в этом.

Формат входного файла

Единственная строка содержит универсальное заклинание, которое открыл Сандро. Заклинание — непустая строка из строчных латинских букв длиной не более 50.

Формат выходного файла

Выведите любое из заклинаний, обладающих, по мнению Сандро, наибольшей силой.

Пример

<code>input.txt</code>	<code>output.txt</code>
tebidoh	tebidoh

Задача С. Тайна происхождения человека проясняется!

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Сенсацией завершился очередной съезд британских учёных, проходивший в городе Баш. Профессор Надувайло Д.В. выступил с докладом, во многом меняющим современное представление о мире.

Ранее считалось, что молекулы ДНК человека и животных состоят из четырёх нуклеотидов, обозначаемых символами «А», «С», «G», «Т». Однако профессор Надувайло доказал, что у каждого нуклеотида существует соответствующий ему антинуклеотид, который также может входить в состав молекулы ДНК. Антинуклеотиды профессор обозначил символами «a», «c», «g», «t», причём буквой «a» он обозначил антинуклеотид для «А», буквой «c» — антинуклеотид для «С» и т. д.

Кроме того, в ходе исследований профессор заметил, что строки ДНК человека имеют особое строение. Профессор считает, что *человеческая* строка ДНК — это либо пустая строка, либо конкатенация двух человеческих строк (то есть одна человеческая строка, приписанная в конец другой), либо человеческая строка, в начало которой добавлен некоторый нуклеотид, а в конец — соответствующий ему антинуклеотид. Например, строки «CcAaGg» и «ACca» являются человеческими, а «cC» и «GTgt» — нет.

Профессор Надувайло предположил, что для установления степени родства человека и животного достаточно лишь проверить, являются ли человеческими некоторые подстроки строки ДНК этого животного. Чтобы разработать эффективный способ проведения таких проверок, коллектив британских учёных пригласил лучших программистов, то есть вас.

Формат входного файла

В первой строке записана строка ДНК некоторого животного, состоящая из символов «А», «С», «G», «Т», «a», «c», «g», «t» и имеющая длину L ($1 \leq L \leq 100\,000$). Во второй строке записано целое число q — количество подстрок данной строки, которые требуется проверить ($1 \leq q \leq 50\,000$). В i -й из следующих q строк через пробел записаны целые числа l_i и r_i — номер первого и последнего символа проверяемой подстроки ($1 \leq l_i \leq r_i \leq L$).

Формат выходного файла

Выведите строку длины q , состоящую из символов «0» и «1». i -й символ должен быть равен единице, если подстрока, начинающаяся в позиции l_i и заканчивающаяся в позиции r_i , является человеческой, и нулю в противном случае.

Пример

input.txt	output.txt
caAgtGTtgAacCc	01101
5	
2 3	
6 9	
10 11	
1 14	
13 14	

Задача D. Аншлаг, аншлаг!

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Осенью в одном из популярных кинотеатров Екатеринбурга все сеансы проходят с аншлагом, поскольку студентам нравится вместо посещения лекций проводить время в уютном кресле за просмотром какого-нибудь фильма. К сожалению, расстояние между рядами в кинозале маленькое, и зрители, пробираясь перед началом фильма к своим местам, вынуждены спотыкаться о ноги уже сидящих. Заходя в зал, зритель думает, с какой стороны ряда он будет пробираться к своему месту (с левой или с правой), и выбирает сторону так, чтобы споткнуться о меньшее число людей. В случае равенства зритель выбирает ту сторону, к которой его место ближе.

Студент философского факультета Вася, ярый любитель кино и столь же ярый ненавистник математики, первым купил билет на очередную премьеру. Когда Вася вошёл в зал и сел на своё место, он увидел, что все остальные кресла в его ряду ещё пусты. Вася точно знал, что к началу сеанса зал заполнится до отказа, а это значило, что с минуты на минуту о его ноги начнут спотыкаться другие кинолюбители, пробирающиеся к своим местам. Несмотря на всю свою нелюбовь к математике, Вася мгновенно оценил, какое максимальное количество человек споткнётся о его ноги, прежде чем все зрители займут свои места. А вы сможете?

Формат входного файла

В единственной строке через пробел записаны целые числа n и k — количество мест в том ряду, где сидит Вася, и номер его места соответственно ($1 \leq k \leq n \leq 50$; n чётно). Места в ряду нумеруются с единицы.

Формат выходного файла

Выведите максимальное количество человек, которое споткнётся о ноги Васи.

Пример

<code>input.txt</code>	<code>output.txt</code>
4 1	1

Задача Е. Кто ходит в гости. . .

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Программный комитет школьных соревнований по программированию, проходящих в УрГУ — многочисленная, весёлая и дружная команда. Дружная настолько, что общения в университете им явно не хватает, поэтому они часто ходят друг к другу в гости. Все ребята в программном комитете очень спортивные и ходят только пешком.

Однажды хранитель традиций олимпиадного движения УрГУ подумал, что на пешие прогулки от дома к дому члены программного комитета тратят слишком много времени, которое могли бы вместо этого потратить на придумывание и подготовку задач. Чтобы доказать это, он решил посчитать, какое расстояние в среднем преодолевают члены комитета, когда ходят друг к другу в гости. Хранитель традиций достал карту Екатеринбурга, нашёл на ней дома всех членов программного комитета и выписал их координаты. Но координат оказалось так много, что хранитель не смог справиться с этой задачей самостоятельно и попросил вас помочь ему.

Город Екатеринбург представляет собой прямоугольник со сторонами, ориентированными по сторонам света. Все улицы города идут строго с запада на восток или с севера на юг, проходя через весь город от края до края. Дома всех членов программного комитета расположены строго на пересечении каких-то двух перпендикулярных улиц. Известно, что все члены комитета ходят только по улицам, поскольку идти по тротуару гораздо приятнее, чем по дворовым тропинкам. И, конечно, при переходе от дома к дому они всегда выбирают кратчайший путь. Программный комитет очень дружный, и все его члены ходят в гости ко всем одинаково часто.

Формат входного файла

Первая строка содержит целое число n — количество членов программного комитета ($2 \leq n \leq 10^5$). В i -й из следующих n строк через пробел записаны целые числа x_i, y_i — координаты дома i -го члена программного комитета ($1 \leq x_i, y_i \leq 10^6$).

Формат выходного файла

Выведите среднее расстояние, которое проходит член программного комитета от своего дома до дома своего товарища, округлённое вниз до целых.

Пример

<code>input.txt</code>	<code>output.txt</code>
3 10 10 20 20 10 20	13

Задача F. Магические числа Знайки

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Знайка — увлекающаяся натура. В данный момент он увлечён тем, что исследует свойства наборов чисел. Знайка выписывает некоторый набор различных целых положительных чисел (Знайка называет его *порождающим набором*), после чего суммирует все записанные цифры, а результат заносит в специальный блокнот. Например, после обработки набора чисел 7, 12, 43 в блокноте Знайки появится число $17 = 7 + 1 + 2 + 4 + 3$. Знайка решил, что результатом такой операции могут быть лишь *магические* числа.

Незнайка смеётся над Знайкой. Он считает, что у любого числа существует порождающий набор, и даже поспорил со Знайкой, что всегда сможет такой набор построить.

Помогите Незнайке выиграть спор и построить какой-нибудь порождающий набор для заданного числа.

Формат входного файла

В единственной строке записано целое число n ($0 < n < 10^5$).

Формат выходного файла

Если для данного n можно построить порождающий набор, в первой строке выведите количество чисел в таком наборе, а во второй строке выведите через пробел сами эти числа. Все числа должны быть целыми, положительными и строго меньшими 10^5 . Кроме того, все числа в наборе должны быть различными. Если существует несколько порождающих наборов, выведите любой из них. Если ни одного порождающего набора не существует, выведите -1 .

Пример

<code>input.txt</code>	<code>output.txt</code>
17	3 7 12 43

Задача G. Проклятие Team.GOV

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Все имена в этой задаче вымышлены, совпадения случайны.

Давным-давно в УрГУ была команда Alarm. Её участники, Лёша, Вадик и Миша, очень любили вместо учёбы ездить на турниры по программированию. В этом составе они выступали целый год. Но однажды, в результате конфликта на турнире в Казани, Лёша и Вадик выгнали Мишу из команды, а её название поменяли на Team.GOV...

Недолго думая, Лёша и Вадик решили, что третьим будет Саша. Но он не пришёл на чемпионат УрГУ, сославшись на неотложные дела. Лёше и Вадиду пришлось участвовать в этом чемпионате вдвоём. Саша всё-таки остался в команде и принял участие в четвертьфинале чемпионата мира, который команда Team.GOV проиграла. Тем не менее, команда «Лёша, Вадик, Саша» внезапно превратилась в команду «Ваня, Лёша, Вадик» и поехала на полуфинал чемпионата мира, который, впрочем, тоже проиграла.

Поговаривают, что команду Team.GOV кто-то проклял. После распада Alarm её состав меняется от турнира к турниру.

Близится очередной четвертьфинал чемпионата мира, и оргкомитет обязал Лёшу с Вадиком найти третьего. Они составили список претендентов в команду и посчитали рейтинг каждого претендента по секретной формуле. Сила команды равна сумме рейтингов её участников. Лёша и Вадик хотят собрать самую сильную команду. Но команда проклята... Волею судьбы, Лёша не придёт на турнир, если на него соберётся состав, который уже когда-то выступал (и тогда сила команды уменьшится на Лёшин рейтинг). Если же даже в этом случае (то есть без Лёши) на турнир соберётся состав, уже выступавший ранее, то команду Team.GOV внезапно дисквалифицируют во время пробного тура, и она вовсе не примет участия в соревновании. Помогите постоянным членам Team.GOV выбрать третьего участника так, чтобы в четвертьфинале выступила команда наибольшей силы.

Формат входного файла

В первой строке записано целое число n ($1 \leq n \leq 100$) — количество турниров, на которых выступала команда Team.GOV. Далее следуют n строк, каждая из которых описывает один турнир. В начале строки записано количество участников, пришедших на турнир (целое число от одного до трёх). Далее через пробел записаны фамилии этих участников — различные непустые строки, состоящие из строчных и прописных латинских букв, длиной не более 50. Фамилии участников команды перечислены в алфавитном порядке. Фамилия Лёши — Efremov, фамилия Вадика — Kantorov. Гарантируется, что во всех составах присутствует Вадик, а во всех составах из трёх человек присутствует Лёша. Все перечисленные составы различны.

В следующей строке через пробел записаны целые числа r_e и r_k ($1 \leq r_e, r_k \leq 666$) — рейтинги Лёши и Вадика, соответственно. Следующая строка содержит целое число m ($1 \leq m \leq 100$) — количество претендентов в команду. В каждой из следующих m строк через пробел записаны фамилия претендента и его рейтинг — целое число от 1 до 666. Фамилии всех претендентов различны. В списке претендентов нет ни Лёши, ни Вадика.

Формат выходного файла

Если при любом выборе третьего участника команда Team.GOV будет дисквалифицирована, выведите единственную строку «Fail». Иначе в первой строке выведите «Win», а во второй — фамилию

претендента, который станет третьим участником Team.GOV. Если возможных ответов несколько, выведите любой.

Примеры

input.txt	output.txt
6 3 Efremov Kantorov Rubinchik 2 Efremov Kantorov 3 Efremov Kantorov Kokovin 3 Burmistrov Efremov Kantorov 3 Efremov Kantorov Pervukhin 2 Kantorov Pervukhin 100 10 6 Fominykh 200 Komarov 34 Pervukhin 250 Golubev 23 Soboleva 50 Gein 50	Win Fominykh
2 3 Efremov Fominykh Kantorov 2 Fominykh Kantorov 99 666 1 Fominykh 100	Fail

Задача Н. Сундук мертвеца

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Капитан Джек Воробей мчался на всех парусах своей «Чёрной Жемчужины» к маленькому островку в Карибском море, на котором был зарыт *сундук мертвеца*. По легенде, в сундуке было спрятано сердце Дейви Джонса, дающее его обладателю безграничную власть и способность повелевать морями. Джек очень спешил, потому что знал, что к этому же острову мчатся его враги, также желающие заполучить столь ценный артефакт.

Когда «Жемчужина» приблизилась к острову, капитан Джек Воробей спустил на воду шлюпку и поплыл к берегу. . . И вот Джек уже пробирается через густые джунгли к своей заветной цели, которая находится в самом центре острова. Не прошло и получаса, как сундук был выкопан и Джек смог насладиться звуком бьющегося сердца Дейви Джонса. Пришла пора сматывать удочки, Джек вскочил и приготовился бежать обратно к шлюпке. И только тут он сообразил, что, охваченный жаждой добычи, он не запомнил, откуда пришёл к этому месту. А значит, он не знал, в каком направлении нужно идти, чтобы вернуться к шлюпке. Джек на мгновение охватила лёгкая паника, ведь с минуты на минуту здесь могли оказаться те, кто тоже хотел заполучить сердце, и тогда. . . В общем, лучше было убраться с острова до их появления.

К счастью, Джек захватил с собой компас и карту острова. С их помощью Джек мог двигаться по любому маршруту на острове. Помогите Джеку выбрать такой маршрут, чтобы он смог увидеть свою шлюпку как можно раньше.

Остров представляет собой круг радиуса r , а сундук был зарыт в самом его центре. Остров полностью покрыт густыми джунглями, поэтому Джек сможет увидеть море, только оказавшись на береговой линии. Шлюпка находится в воде вблизи берега, так что её видно только с тех точек берега, которые отстоят от точки берега, ближайшей к шлюпке, не более чем на a градусов. Также известно, что Джек не сможет увидеть корабль до того, как увидит шлюпку.

Формат входного файла

В единственной строке через пробел записаны вещественные числа r и a ($1 \leq r \leq 1000$; $0.001 \leq a \leq 45$). Числа содержат не более трёх знаков после десятичной точки.

Формат выходного файла

Выведите с точностью не менее шести знаков после десятичной точки расстояние, которое в худшем случае придётся пройти Джеку до того, как он увидит шлюпку, при условии выбора им оптимального маршрута.

Пример

<code>input.txt</code>	<code>output.txt</code>
1.000 30.000	6.0000000

Задача I. Челмедведосвин

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

В Саус-Парке было раннее зимнее утро. Взрослые привычно спешили на работу, а дети — в школу. Закадычные друзья Стэн, Кайл, Кенни и Эрик стояли на остановке и ждали школьный автобус. День обещал быть длинным и скучным.

Однако в школе ребят ждала встреча с необычным человеком. Всё началось с того, что школьный психолог собрал всех детей в спортзале и объявил: «Так, дети! Внимание! Сегодня в гостях у нас важная персона. Кто назовёт мне имя нашего предпоследнего вице-президента?»

Конечно же, никто из детей не знал, что этого человека звали Эл Гор. И, как выяснилось чуть позже, возможно, и к лучшему.

Он вышел на трибуну и сказал следующее: «Спасибо, Мистер Маки, здравствуй, начальная школа Саус-Парка. Я хочу рассказать о самой страшной угрозе для нашей планеты. Она может положить конец человеческой расе. Я говорю о челмедведосвине. Это создание в одиночку наводит страх на Землю. Это наполовину человек, наполовину медведь, наполовину свинья. Многие считают, что челмедведосвина не существует, но уверяю вас, что он существует. Я серьёзно. Ему безразлично, кто вы и чем занимаетесь. Он хочет до вас добраться. Но не бойтесь, потому что я вас спасу. И когда-нибудь потом, когда мир будет свободен от челмедведосвина, все скажут мне спасибо».

Конечно же, никто не придавал значения его словам. Все просто пожали плечами и разошлись на уроки. Но вечером Эрик вдруг понял, что Эл Гор сказал полнейшую чепуху.

Сами посудите, как может быть в одном существе три половины других? Эрик сказал это Кенни, но тот лишь отмахнулся, сказав, что Эл Гор подразумевал по трети каждого из животных. Но Эрик был прав. Действительно, не существует такой родословной, чтобы в челмедведосвине было поровну свиньи, медведя и человека.

Родословная существа — это последовательность скрещиваний существ, приводящая к его появлению. В результате скрещивания существа A и существа B образуется новое существо, наполовину состоящее из A и B . Если в существе A было a процентов некоторого животного, а в существе B — b процентов этого же животного, то в получившемся при их скрещивании существе будет $\frac{a+b}{2}$ процентов данного животного.

Теперь Эрик хочет научиться составлять возможную родословную существа по данным о том, из каких животных и в каких соотношениях оно состоит.

Формат входного файла

Первая строка содержит название существа. Название состоит из строчных латинских букв и дефисов и имеет длину от 1 до 200. Дефисы отделяют друг от друга непустые названия животных. Каждое животное встречается в названии существа не более одного раза. Существо состоит не менее чем из одного и не более чем из десяти различных животных. Во второй строке через пробел записаны дроби вида a/b ($1 \leq a \leq b \leq 1500$), i -я дробь в этом списке указывает, какую часть существа составляет i -е животное в его названии. Сумма всех дробей равна единице.

Формат выходного файла

Выведите родословную существа в виде последовательного описания того, каким образом появились на свет его предки и оно само. В первой строке выведите число n — количество строк в родословной ($1 \leq n \leq 10\,000$). Далее выведите $n - 1$ строку, k -я из которых должна содержать описание того, что из себя представляет k -й предок существа. Если k -й предок является животным, то строка должна содержать только название этого животного. Если же он был получен скрещиванием двух ранее описанных предков, то строка должна содержать пару целых чисел в диапазоне от 1 до $k - 1$ — номера строк родословной, в которых описаны эти предки. В последней строке выведите

в том же формате, каким образом появилось на свет существо, описанное во входных данных. Все описанные предки существа должны участвовать хотя бы в одном скрещивании. Если возможных родословных несколько, выведите любую из них. Если требуемую родословную построить нельзя, выведите строку «No solution».

Примеры

input.txt	output.txt
man-bear-pig 1/2 1/4 1/4	7 man pig 1 2 bear man 4 5 3 6
man 1/1	1 man
man-bear-pig 1/3 1/3 1/3	No solution

Задача J. Укроп

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Иван Васильевич — заядлый дачник. До него дошли слухи о пользе и красоте укропа, поэтому весной он засеял весь участок двумя сортами этого растения.

Когда укроп вырос, Иван Васильевич заготовил n ящиков укропа первого сорта и m ящиков укропа второго сорта. Масса каждого ящика выражена целым числом килограммов, причём массы всех ящиков различны. Чтобы порадовать своих соседей — Ивана Ивановича и Ивана Никифоровича, Иван Васильевич решил подарить каждому из них по ящику укропа каждого сорта. Чтобы Иван Иванович не поссорился с Иваном Никифоровичем, суммарные массы двух ящиков, подаренных каждому из соседей, должны совпадать. Перебрав все возможные варианты, Иван Васильевич понял, что это сделать невозможно. Определите массы ящиков с укропом, заготовленных Иваном Васильевичем.

Формат входного файла

В единственной строке через пробел записаны целые числа n и m ($2 \leq n, m \leq 50$).

Формат выходного файла

В первой строке выведите через пробел n чисел — массы ящиков с первым сортом укропа. Во второй строке выведите через пробел m чисел — массы ящиков со вторым сортом укропа. Массы должны быть положительными целыми числами, не превосходящими 10^9 . Кроме того, все массы должны быть различными. Если задача имеет несколько решений, выведите любое из них. Если задача не имеет решения, выведите строку «It is a lie!».

Пример

<code>input.txt</code>	<code>output.txt</code>
2 3	1 2 5 10 12