

Задача А. Страшное марсианское слово

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

У жителей разных планет разные суеверия. Марсиане смеются над землянами, страдающими трискайдекафобией и гексакосиогексеконтагексафобией. А земляне подшучивают над тем, что в марсианском языке есть страшное слово. Марсиане боятся не только этого страшного слова, но и всех слов, которые получаются перестановкой букв страшного слова.

В марсианском алфавите 729 000 букв. Журналист Овчинников, живущий на Марсе и изучающий язык марсиан, записывает буквы марсианского алфавита тройками символов с ASCII кодами от 33 до 122. Недавно он написал книгу о быте и культуре марсиан. Перед тем как отправить книгу в печать, Овчинников хочет посчитать количество подстрок в тексте книги, которые напугают марсиан.

Формат входных данных

В первой строке записано страшное марсианское слово, состоящее не более чем из 8 000 марсианских букв. Во второй строке приводится текст книги Овчинникова, длина которого не превосходит 500 000 марсианских букв. И в страшном слове, и в тексте книги есть хотя бы одна буква. Каждая марсианская буква записана тройкой символов с ASCII кодами от 33 до 122, соседние буквы разделены пробелом.

Формат выходных данных

Выведите количество подстрок в тексте книги журналиста Овчинникова, являющихся страшными для марсиан.

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
aaa bbb ccc aaa aaa bbb ccc aaa zzz aaa bbb ccc	3

Страшными для марсиан являются две подстроки «aaa bbb ccc» (начинающиеся со второй и с седьмой позиции в тексте) и подстрока «bbb ccc aaa».

Задача В. Орбитальная атака

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Боевой космический корабль «Рикенбакер» приближался к планете Оркут, последнему оплоту враждебной расы подан. Преимущество было на стороне «Рикенбакера» — весь космический флот подан уже был уничтожен. Но тут поступили пугающие разведданные — на поверхности Оркута было расположено несколько пусковых установок с ракетами класса «оркут-космос».

Все пусковые установки расположены на небольшом плоском плацдарме на поверхности Оркута. «Рикенбакер» оснащён дальнобойным лазером, способным поразить пусковую установку раньше, чем корабль приблизится к планете на опасное расстояние. С системой наведения лазера связана прямоугольная декартова система координат. К сожалению, лазер может поразить только такую цель, обе координаты которой в этой системе координат являются целыми.

Капитану «Рикенбакера» сообщили точные координаты каждой пусковой установки. Теперь он хочет перед началом стрельбы один раз перенастроить лазер, переместив начало системы координат в другую точку плоскости, чтобы лазер смог поразить наибольшее число пусковых установок. При этом капитан не может повернуть оси координат системы наведения.

Помогите капитану выбрать точку, в которую нужно перенести начало системы координат. Если существует несколько таких точек, выберите ту, расстояние от которой до прежнего начала системы координат минимально.

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 50\,000$) — количество пусковых установок. В следующих n строках записаны координаты этих установок — вещественные числа, не превосходящие 100 по абсолютной величине и имеющие не более трёх знаков после десятичной точки. Координаты разных установок могут совпадать.

Формат выходных данных

Выведите два числа — максимальное число установок, которое сможет уничтожить лазер на «Рикенбакере», и минимальное расстояние, на которое потребуется для этого перенести начало системы координат. Расстояние требуется вывести с абсолютной погрешностью не более 10^{-5} .

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
3 0.500 0.200 0.500 0.500 -0.500 0.200	2 0.53852
2 1.000 1.000 1.000 1.000	2 0.00000

Задача С. Космический пьяница

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Мода на карточные игры переменчива. Ещё два галактических года назад никто и не слышал о боге забытой планете Техас и об игре «пьяница», которую так любят местные жители. Сейчас же вы вряд ли найдёте хоть одно казино галактики, в котором не просаживают состояния в эту незамысловатую игру.

Галактический комитет по карточным играм определил правила «пьяницы» следующим образом. Игра ведётся двумя одинаковыми колодами из n карт, все карты колоды имеют различные достоинства, занумерованные целыми числами от 1 до n . Каждый из двух игроков в начале игры получает полную колоду и тщательно её тасует. Колода кладётся на игровой стол рубашкой вверх. Затем оба игрока одновременно открывают верхнюю карту своей колоды. Если достоинства этих карт совпадают, то обе карты откладываются и не участвуют в дальнейшей игре. В противном случае игрок, выложивший карту большего достоинства, забирает обе карты и подкладывает их под свою колоду так, что карта большего достоинства оказывается самой нижней, а меньшего достоинства — второй снизу. После этого игроки снова открывают верхнюю карту своей колоды. Игрок, оставшийся без карт, проигрывает. Если карты закончились у игроков одновременно, объявляется ничья.

Шулер Дастин недавно вживил в свой глаз крошечный сканер, который позволяет ему видеть порядок карт в колоде соперника. Теперь он хочет перетасовать свои карты так, чтобы гарантировать себе победу. Конечно, если Дастин сложит карты в своей колоде в том же порядке, что и у соперника, то он легко добьётся ничьей. Но это Дастина не интересует — ему нужна только победа.

Формат входных данных

В первой строке записано целое число n ($2 \leq n \leq 500$) — количество карт в колоде каждого игрока. Во второй строке записана перестановка чисел от 1 до n — достоинства карт в колоде соперника Дастина в порядке сверху вниз.

Формат выходных данных

Выведите в первой строке слово «YES», если Дастин сможет расположить свои карты в порядке, который гарантирует ему победу. В этом случае выведите во второй строке искомый порядок карт в колоде Дастина сверху вниз. Если Дастин не сможет гарантировать себе победу, выведите единственное слово «NO».

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
4 2 3 4 1	YES 2 4 1 3

Задача D. Досье орбитальной атаки

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дальнобойный лазер, установленный на борту боевого космического корабля «Рикенбакер», успешно уничтожил пусковые установки на поверхности враждебной планеты Оркут. Не прошло и галактических суток, как раса шодан сдалась.

Капитан корабля «Рикенбакер» рассчитывал на повышение в звании. Его завистники же старались убедить высшее командование в том, что капитан лжёт — никому не под силу так быстро и успешно сориентироваться в ситуации и поразить так много целей во время орбитальной атаки. Капитан понял, что ему предстоит защитить свою честь и подготовить подробный доклад о проведённой операции. Для начала он решил изобразить все уничтоженные установки на одном графике.

С системой наведения лазера связана прямоугольная декартова система координат. Все координаты уничтоженных установок являются в этой системе координат целыми. Оси координат необходимо изобразить с помощью символов «|» (вертикальная черта, для изображения оси ординат), «-» (минус, для изображения оси абсцисс), «+» (плюс, для изображения начала координат). Точки, в которых находились уничтоженные установки, нужно изобразить символом «*» (звёздочка). Все остальные точки нужно изобразить символом «.» (точка). Ось абсцисс на графике должна быть направлена вправо, а ось ординат — вверх. Один символ на графике соответствует единице по оси абсцисс по горизонтали и единице по оси ординат по вертикали. Оси координат обязательно должны присутствовать на графике, но они могут оказаться полностью закрыты символами «*».

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 250$) — количество уничтоженных пусковых установок. В каждой из следующих n строк записаны координаты очередной установки. Все координаты целые и не превосходят 100 по абсолютной величине. Никакие две установки не расположены в одной точке.

Формат выходных данных

Выведите требуемый график. Первая строка должна соответствовать максимальному значению y (или 0), а последняя — минимальному значению y (или 0). В каждой строке должно быть записано одинаковое количество символов. Первый символ строки должен соответствовать минимальному значению x (или 0), а последний — максимальному значению x (или 0).

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
8	*.....
-10 5	.*.....
-7 3	...*....
-4 2*..
-9 4*.....
0 1	-----+--*-----
6 -1*..
3 0
8 -3*

Задача Е. Многорукие братья

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Москиты, завезённые на Уран с Земли, мутировали и расплодились в больших количествах. Теперь разумным жителям Урана приходится постоянно плётать самих себя, убивая насекомых, нороящих высосать из них протоплазму. Возможно, именно поэтому количество рук у новых отпочковавшихся уранцев может быть больше, чем у их родителя. У первого ребёнка каждого уранца столько же рук, сколько у его отца. У второго ребёнка — рук в два раза больше, чем у отца, у третьего — в три раза больше, и так далее. Ни один уранец за свою жизнь не может отпочковать больше g детей.

А теперь о кино. Премьера 146-го эпизода «Звёздных войн» пройдёт одновременно в g кинозалах, а в очереди за билетами выстроились уже n уранцев. Кассир Федя боится, что братья-уранцы во время просмотра фильма начнут выяснять отношения друг с другом. Поэтому он хочет продать билеты так, чтобы два брата не попали в один кинозал. Конечно, он не в курсе родственных связей зрителей, но зато прекрасно видит, сколько рук у каждого из них. Сможет ли Федя по этой информации распределить уранцев по залам так, чтобы ни в каком зале гарантированно не оказалось двух уранцев, имеющих общего отца?

Формат входных данных

В первой строке записаны целые числа n и g ($1 \leq n \leq 50\,000$; $2 \leq g \leq 5$). Далее записаны n попарно различных целых чисел, задающих количество рук у уранцев, стоящих в очереди. У каждого уранца не менее одной и не более 10^9 рук.

Формат выходных данных

Если Феде удастся распределить уранцев по кинозалам требуемым образом, выведите в первой строке «Yes», а затем выведите n чисел, указывающих номер зала для каждого зрителя. Зрителей нужно описывать в том же порядке, в котором они заданы на входе. Кинозалы занумерованы числами от 1 до g . Если задачу решить нельзя, выведите «No».

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
5 2	Yes
1	1
2	2
3	1
4	1
6	2

Задача F. Китайский хоккей 3

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

В высшей лиге чемпионата КНР по настольному хоккею участвуют n лучших игроков со всех уголков галактики, колонизированных республикой. В течение сезона каждый участник лиги должен сыграть с каждым ровно один матч. Игра ведётся до тех пор, пока не будут забиты ровно 3 гола. Таким образом, в настольном хоккее не бывает ничьих — все матчи заканчиваются либо со счётом 3:0, либо со счётом 2:1 в пользу одного из игроков. В итоговой таблице результатов лиги игроки упорядочиваются по суммарному количеству забитых ими во всех встречах голов.

Вы работаете в букмекерском агентстве, которое принимает прогнозы на итоговый вид таблицы результатов. Для победы в конкурсе прогнозов необходимо абсолютно точно назвать количество голов, забитых победителем, количество голов, забитых игроком, занявшим второе место, и так далее, вплоть до количества голов, забитых игроком, занявшим последнее место. При этом угадывать имена игроков не нужно. Помогите агентству посчитать количество всевозможных прогнозов.

Формат входных данных

В единственной строке записано целое число n ($2 \leq n \leq 50$).

Формат выходных данных

Выведите количество возможных прогнозов на итоговые результаты, взятое по модулю $10^9 + 7$.

Пример

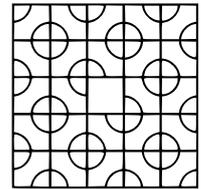
<i>Входные данные</i>	<i>Выходные данные</i>
3	8

Для трёх игроков возможны следующие прогнозы: (6, 3, 0), (6, 2, 1), (5, 4, 0), (5, 3, 1), (5, 2, 2), (4, 4, 1), (4, 3, 2), (3, 3, 3).

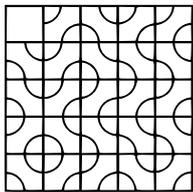
Задача G. Меандр

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Робот-гроссмейстер E2E4 с позором уступил в 1/128 финала чемпионата галактики по игре «меандр». Он практически довёл партию до победы, но совершил неслыханную ошибку — просто не увидел победный ход! Окрылённый этой ошибкой соперник воспользовался шансом и выиграл. Похоже, что модуль искусственного интеллекта E2E4 начал сбоить. Можете ли вы быстро помочь роботу, сделав так, чтобы он больше не проигрывал партии в «меандр» столь обидным образом?



Для игры «меандр» используют 24 квадратные плитки, которые выкладываются на игровое поле, состоящее из 25 клеток, так, чтобы одна клетка осталась свободной. Если смотреть на поле сверху, то на двенадцати плитках изображён рисунок , а на других двенадцати — рисунок . Один из вариантов начального положения изображён на рисунке справа сверху.



Игроки ходят по очереди. За один ход игрок может одновременно сдвинуть одну, две, три или четыре плитки вдоль одной из сторон игрового поля. Плитки перемещаются по поверхности доски, поворачивать их нельзя. Игра продолжается до тех пор, пока один из игроков не выиграет, сложив узор, в котором по крайней мере три плитки образуют непрерывную линию, соединяющую два края доски (либо противоположные, либо смежные, сходящиеся в одном углу). Такой узор показан на рисунке слева (линия соединяет левый и нижний края доски).

Напишите программу, которая определяет, является ли заданный узор победным, и, если нет, то можно ли получить из него победный узор за один ход.

Формат входных данных

Входные данные представляют собой пять строк по пять символов в каждой строке. Символом «*» обозначается свободная клетка. Символом «/» обозначается плитка , а символом «\» — плитка . Гарантируется, что входные данные содержат ровно один символ «*» и по двенадцать символов «/» и «\».

Формат выходных данных

Выведите «WIN», если узор уже является победным. Если узор не является победным, но есть ход, переводящий его в победный, выведите этот ход. Ход описывается двумя символами: первый задаёт направление движения («L» — влево, «R» — вправо, «D» — вниз, «U» — вверх), а второй — количество сдвигаемых плиток (от 1 до 4). Например, «U2» задаёт ход «сдвинуть две плитки вверх». Если существует несколько ходов, переводящих заданный узор в победный, выведите любой из них. Если узор не является победным и его нельзя перевести в победный за один ход, выведите единственную строку «FAIL».

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
*\\// \\\\/ \\/// /\\\\ \\//	WIN
\\// \\// //* \\\\/ \\//	D2

Задача Н. Робот на прямой

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

На планете Т4-Э7 звёздной системы Тау Кита активно развивается робототехника. Недавно профессор Бобов продемонстрировал свою новую разработку — автономного робота, который способен свободно перемещаться по прямой. Программа робота принимает на вход действительное число x , которое полностью задаёт поведение робота в будущем. В первую секунду робот перемещается на расстояние $f(x) = ax^2 + bx + c$ вдоль прямой (если $f(x)$ положительно, то он перемещается вправо, а если отрицательно — влево). Во вторую секунду робот перемещается уже на расстояние $f(x + 1)$, в третью — на $f(x + 2)$ и так далее.

Для презентации Бобов захотел добиться того, чтобы робот через k секунд после начала работы программы вернулся в ту же точку, в которой он находился исходно. И тут возник вопрос: а что если подходящего x просто не существует? Ведь может оказаться так, что как ни задавай входной параметр, робот не сможет по истечении k секунд вернуться в исходную точку. Помогите профессору Бобову — найдите минимальное k , при котором такое может случиться.

Формат входных данных

В первой строке записано целое число t ($1 \leq t \leq 1000$) — количество тестов. В каждой из следующих t строк записан очередной тест — целые числа a, b, c , характеризующие поведение робота. Числа a, b, c не превосходят 10^9 по абсолютному значению. Число a положительно.

Формат выходных данных

Для каждого теста выведите минимальное целое положительное число k , для которого не существует параметра x , возвращающего робота на место через k секунд. Если такого k не существует или оно больше 10^{18} , выведите «Infinity».

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
2	2
1 -2 1	9
1 1 -6	

Задача I. Лучший фильм галактики

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Профессор Валентин Владимирович, член галактической киноакадемии, завтра примет участие в голосовании по номинации «лучший фильм». Но профессор ещё не смотрел ни одного из n фильмов-номинантов, да и сегодня у него совсем нет времени. Он решил попросить о помощи k своих студентов, которые до сих пор не получили у него зачёт. Если студенты посмотрят все фильмы, а потом кратко расскажут о своих впечатлениях, то этой информации будет вполне достаточно, чтобы сделать выбор. . .

Каждый из фильмов-номинантов показывают в кинотеатре только один раз — i -й фильм начинается в момент времени a_i и заканчивается в момент времени b_i . Студент может посмотреть за день сколько угодно фильмов, но любой из них он должен посмотреть полностью от начала и до конца. Студент не может смотреть два фильма одновременно. Известно также, что в i -м фильме снимается c_i красивых актрис. Студент откажется идти на фильм, если в нём будет меньше красивых актрис, чем в предыдущем просмотренном им фильме.

Валентин Владимирович дал студентам подробные указания на тот случай, если они не успеют посмотреть за день все фильмы. Фильмы занумерованы в порядке убывания известности их режиссёров, фильмы известных режиссёров интересуют профессора больше. Если студенты могут посмотреть за день набор фильмов A либо набор фильмов B , то набор A предпочтительнее, если существует такое i , что:

- фильмы с номерами от 1 до $i - 1$ либо содержатся и в A , и в B , либо не содержатся ни в A , ни в B ;
- фильм с номером i содержится в A и не содержится в B .

Помогите Валентину Владимировичу и его студентам найти самый предпочтительный набор фильмов.

Формат входных данных

В первой строке записаны целые числа n и k ($1 \leq n \leq 100$; $1 \leq k \leq n$). В i -й из n следующих строк записаны целые числа a_i, b_i, c_i ($0 \leq a_i < b_i \leq 86\,399$; $1 \leq c_i \leq 10\,000$).

Формат выходных данных

Выведите строку из n цифр, задающую самый предпочтительный набор фильмов. На i -й позиции должна стоять единица, если i -й фильм входит в набор, и ноль в противном случае.

Примеры

<i>Входные данные</i>	<i>Выходные данные</i>
4 2 1 5 10 3 6 20 5 10 9 5 10 10	1101
4 1 1 5 10 3 6 20 5 10 9 5 10 10	1001

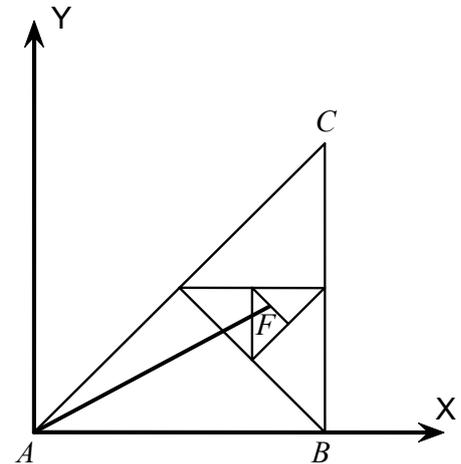
Задача J. Дачи на Марсе

Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Алексей Иванович решил продать свой дачный участок на Марсе и вернуться на родную Венеру. Его участок имеет форму плоского прямоугольного треугольника ABC , на котором вырыт узкий прямой ирригационный канал AF . Увы, Алексею Ивановичу не удалось быстро найти покупателя — оказалось, что современные дачники покупают только участки, на которых есть «сквозной» ирригационный канал, то есть, канал, имеющий ровно две общие точки с линией границы участка.

Поразмыслив, Алексей Иванович понял, что можно разделить свой участок на участки меньшего размера и продать их по отдельности. Он решил действовать следующим образом: поделить исходный участок на два одинаковых треугольных участка такой же формы, проведя высоту к гипотенузе. Затем перейти к половине, внутри которой будет находиться точка F , и проделать с ней то же самое — снова провести высоту к гипотенузе и перейти к меньшему участку, внутри которого будет находиться точка F . Процесс будет продолжаться до тех пор, пока точка F не окажется на очередной высоте, или пока размер новых участков не станет пренебрежимо малым.

Помогите Алексею Ивановичу посчитать суммарную площадь всех получившихся участков, которые ему удастся продать.



Формат входных данных

Введём систему координат так, что точка A будет иметь координаты $(0, 0)$, точка B — координаты $(10, 0)$, а точка C — координаты $(10, 10)$. В единственной строке записаны координаты x и y точки F ($0 < y < x < 10$). Числа x и y заданы не более чем с тремя знаками после десятичной точки.

Формат выходных данных

Выведите суммарную площадь участков, пригодных для продажи, с абсолютной или относительной погрешностью не более 10^{-3} .

Пример

<i>Входные данные</i>	<i>Выходные данные</i>
8.125 4.375	29.6875