

Задачи финального этапа

“Кубка Тамтэк” 2013

Задача А. “N-мерные шарики”

Time limit: 1 sec

Memory limit: 512 M

Как вы уже знаете, сотрудники Компании “Тамтэк” любят в перерывах между работой играть в разные игры. Одна из их любимых игр — N-мерные шарики.

Участвуют два игрока. Правила игры такие: игроки по очереди расставляют n-мерные шарики диаметром d на n-мерном (прямоугольном) поле с вещественными размерами $(x_1, x_2, x_3, \dots, x_n)$, и, тот, кто первый не сможет поставить туда шарик - проиграл, а его противник — выиграл.

На шарики не действуют никакие силы, они остаются там, куда их поместили. Шарик можно поместить в любую такую точку стола, где он не пересекается со стенками или другими шариками. Касаться других шаров или стенок можно.

Игра настолько популярна, что руководство компании решило закупить еще t наборов этой игры.

Однажды стажеру Коле пришла в голову идея, что должна существовать какая-то оптимальная стратегия игры, и он хочет узнать, какой игрок будет выигрывать при правильной игре на каждом из новых столов. Для этого самый правильный путь - написать программу, которая сможет найти ответ. Но сам написать такую программу он еще не может, поэтому просит написать Вас.

Ввод:

Тестовый пример состоит из нескольких наборов. Наборы независимы друг от друга.

В первой строке расположено единственное число t - количество новых игровых наборов ($1 \leq t \leq 100$).

Далее идет описание игровых наборов, каждый набор располагается в двух строках: в первой строке целое число d - диаметр шарика ($1 \leq d \leq 30$), во второй строке сначала идет число n - размерность пространства ($2 \leq n \leq 100$) и n чисел x_i -

размеры стола ($1 \leq x_i \leq 100$)

Вывод:

Для каждого набора в новой строке выведите First или Second, в зависимости от того, какой игрок выиграет при правильной игре.

Пример:

Ввод	Вывод
3	First
1	First
2 5 3	Second
3	
4 3 3 4 3	
7	
3 7 6 7	

Первая группа, тест из примера - 0 баллов.

Вторая группа, тесты 2 - 7, $t \leq 5$, n в {2, 3, 4}, 30 баллов

Третья группа, тесты 8 - 13, $t \leq 100$, n в {2, 3, 4}, 30 баллов

Четвертая группа, тесты 14 - 21, $t \leq 100$, $n \leq 100$, 40 баллов.

Задача В. “Чат”

Time limit: 1 sec

Memory limit: 512 M

Ровно в полночь в чат вошли n человек. Каждый человек мог писать сообщения в чат и видел все сообщения, которые были написаны во время его пребывания в чате. Пробыв некоторое время в чате, каждый человек вышел из него.

Для i -го человека известно количество сообщений j -го пользователя $a_{i,j}$, которое он видел. При этом неизвестно, сколько своих сообщений он видел, то есть $a_{i,i}$ не определено.

Участников чата заинтересовал вопрос: в каком порядке они покинули чат?

Требуется установить последовательность, в которой участники могли покинуть чат, и, если есть несколько возможностей, вывести минимальную в лексикографическом порядке. Последовательность a_1, a_2, \dots, a_n меньше в лексикографическом порядке последовательности b_1, b_2, \dots, b_n , если для некоторого $i > 0$ выполняется: $a_1 = b_1, a_2 = b_2, \dots, a_i = b_i, a_{i+1} < b_{i+1}$.

Ввод:

На ввод подается один или несколько тестов.

Первая строка описания каждого теста содержит число n ($0 < n \leq 1200$). (Значение $n == 0$ означает конец набора тестов). Для любого другого n далее следует n строк, каждая из которых содержит n целых чисел --- $a_{i,j}$ ($0 \leq a_{i,j} \leq 10\,000$). Значение $a_{i,i}$ может быть любым, так как по условию оно считается неопределенным и не должно использоваться.

Размер входного файла не превышает 10 мегабайт.

Вывод:

Для каждого теста выведите в выходной файл одну строку. Если ответ существует, то выведите в этой строке n целых чисел --- последовательность, в которой пользователи могли покинуть чат. Если возможных ответов несколько, выведите лексикографически минимальный. Если допустимых ответов нет, выведите -1.

Пример:

Ввод	Вывод
3 0 1 1 2 0 0 3 1 0 3 0 1 1 4 0 0 3 1 0 0	2 1 3 -1

Задача С. “Массовый зачет”

Time limit: 1 sec

Memory limit: 512 M

Одажды стажеры стояли в очереди в столовой и стажер Коля стал рассказывать, как тяжело стало сдавать зачеты и экзамены. Коллеги его стали подбадривать и Дима рассказал историю, как он сдавал экзамен по “Теории вероятностей”.

Их преподаватель устроил им балльную систему, по которой выставлялся допуск к зачету. При этом положительные баллы начисляются, если сдаешь домашние работы, контрольные и призовые задачи, а все остальные “достижения” - прогул, опоздание, не выполненная домашняя работа и т.д. приносят отрицательные баллы.

Мало того, на зачет преподаватель выставил N задач, за которые можно заработать a_1, a_2, \dots, a_n баллов с вероятностями p_1, p_2, \dots, p_n соответственно, и зачет получал только тот, кто мог набрать не менее половины всех баллов.

Эта история всех стоящих в очереди развеселила, а Коля погрустнел. Ведь его группе предстояло сдавать зачет именно этому преподавателю...

После обеда Коля решил написать программу, которая посчитает его шансы и шансы одногруппников на сдачу зачета, ведь он знал количество одногруппников, список из N задач для каждого, баллы за их задачи и вероятности, с которыми они эти задачи могут решить, но ... программа не работала. Поэтому он обратился к Вам за помощью в решении данной задачи.

Ввод:

В первой строке располагается единственное число T ($1 \leq T \leq 100$) - количество одногруппников Коли.

Далее идет описание набора задач для каждого из одногруппников, по три строки на каждого:

в первой строке располагается единственное число N ($1 \leq N \leq 100$) - количество задач очередного одногруппника;

во второй строке расположено N чисел, a_1, a_2, \dots, a_n ($1 \leq a_i \leq 100$) - количество баллов за соответствующую задачу данного одногруппника;

в третьей строке расположено N чисел p_1, p_2, \dots, p_n ($0 \leq p_i \leq 100$) - вероятности, заданные в процентах, с которыми данный одногруппник может решить соответствующую задачу.

Вывод:

Для каждого одногруппника в отдельной строке выведите его вероятность сдачи зачета. Ответ вывести с точностью не менее 6 десятичных знаков после запятой.

Пример:

Ввод	Вывод
2	0.5000000
5	0.8000000
1 2 3 4 4	
0 100 100 0 50	
2	
5 5	
50 60	

Задача D. “Дешифровка”

Time limit: 1 sec

Memory limit: 512 M

Известный способ шифрования заключается в следующем. Текст представляется в виде последовательности ASCII-кодов своих символов: t_0, t_1, \dots, t_{n-1} . Выбирается секретный ключ k , состоящий из m байт: k_0, k_1, \dots, k_{m-1} . После этого для шифрования вычисляется значение операции хог последовательных элементов ключа и текста: $c_i = t_i \text{ xor } k_{i \bmod m}$. Получившаяся последовательность c_0, c_1, \dots, c_{n-1} является зашифрованным текстом.

Задан зашифрованный текст, найдите исходный текст, если известно, что он представляет собой литературный текст на английском языке.

Ввод:

На вход подается одна текстовая строка, в которой через запятую приведены байты зашифрованного текста. Размер строки не превышает 400000 символов, в ней не менее 5000 и не более 100000 элементов. Длина ключа не больше 20.

Вывод:

Необходимо вывести исходный текст размером ровно n байт.

Пример:

Для тестирования вашей программы вам предоставлен текст размером около 100 килобайт, а также этот же текст, зашифрованный с ключом “password”. Эти тексты можно скачать здесь - <https://omsk-olymp.ru/ejudge/problems/files/handout.zip>.

Задача E. “Вся сила в менеджменте... и печеньках!!!”

Time limit: 1 sec

Memory limit: 512 M

В компании “Татмтэк” решил начать работу над новым мега-проектом и для него уже выделили команду из N программистов и менеджера Микола. Задача Микола состоит в том, чтобы руководить работой программистов так, чтобы достичь наилучших результатов. Для этого существует множество различных методик.

Микола хочет попробовать в этом проекте метод параллельного программирования, так как этот метод в последнее время приобретает большую популярность, и о нем много хороших отзывов.

Этот метод заключается в том, что в любой момент времени код пишет только один программист, а другие наблюдают за его работой. Этот метод очень эффективен, поскольку наблюдающие программисты видят все ошибки, которые делает пишущий программист, и, когда приходит их очередь писать код, они уже знают, что будут писать, как и каким образом они будут избегать этих ошибок. При этом каждый программист должен написать хотя бы q строк кода, иначе все его размышления не приведут к какому-то результату.

Программисты компании “Татмтэк” настолько суровы, что только пишут код, не стирая его. При этом программисты не обязаны писать целое число строк. Например: один программист может написать 10 строк и еще треть строки, а второй допишет оставшиеся две трети и напишет еще 5 строк.

Эффективность команды зависит не только от организации процесса разработки, но и от эффективности каждого из ее членов. Для ее повышения также существуют различные методы. Например, если давать программистам печеньки, то они будут работать гораздо быстрее.

Микола знает, что без печенек i -й программист пишет в день s_i строк кода, с печеньками - f_i , поэтому решил, что разработчики ни в коем случае не должны испытывать недостатка в их количестве, однако с поставщиком ему договориться еще не удалось.

Проект очень большой -- целых L строк кода, а время поджимает, поэтому Миколе срочно нужно распределить работу между программистами. Поэтому он решил распределить ее так, что если у программистов будут печеньки, время на разработку проекта окажется минимальным, а если ему не удастся договориться с поставщиком, то разработка хотя бы уложится в дедлайн (то есть, завершилась бы меньше, чем за d дней).

Так как он сейчас занят переговорами, то просит вас помочь ему в этом.

Ввод:

В первой строке содержится четыре целых числа -- N, L, q, d ($1 \leq n \leq 50000$, $1 \leq L \leq 2 \cdot 10^7$, $1 \leq q \leq 500$, $1 \leq d \leq 2 \cdot 10^7$)

В следующих N строках содержится по 2 числа -- s_i и f_i ($1 \leq s_i < f_i \leq 1000$)

Вывод:

Мы не просим Вас сказать, как нужно распределять работу между программистами, просто скажите, каково будет время разработки, если Миколу удастся договориться с поставщиком. Ответ вывести с точностью не менее 6 десятичных знаков после запятой.

Пример:

Ввод	Вывод
3 1200 100 5 180 900 300 450 200 300	2.361111

Комментарий к тесту:

Такое время можно получить, если первый программист напишет примерно 375 строк кода, второй -- примерно 725, а третий -- 100.

Тесты к этой задаче состоят из четырех групп:

Тест 1. Тест из условия. Оценивается в 0 баллов.

Тесты 2-7. В тестах этой группы $n = 5$. Оцениваются в 30 баллов.

Тесты 8-15. В тестах этой группы $n \leq 100$. Оцениваются в 40 баллов.

Тесты 16-21. В тестах этой группы дополнительные ограничения отсутствуют. Оцениваются в 30 баллов.