

## Задача А. The Friends

Имя входного файла: `standard input`  
Имя выходного файла: `standard output`  
Ограничение по времени: 2 seconds  
Ограничение по памяти: 256 Mebibytes

Джон и Брюс — лучшие друзья. Они почти всё время проводят вместе, играя в игры, смотря телевизор, слушая музыку, решая задачки по программированию и так далее. Однако, похоже, в последнее время между ними случился конфликт из-за знакомой девушки...

*John: We are still the best friends, aren't we?*

*Brus: Okay, I'll give you a chance!*

Возникший конфликт друзья постановили решить жеребьёвкой.

Джон случайным образом выбирает целое число  $J$  в промежутке между  $J_1$  и  $J_2$  включительно. Аналогично, Брюс случайным образом выбирает целое число  $B$  в промежутке между  $B_1$  и  $B_2$  включительно.

Затем, если  $J^B$  ( $J$  в степени  $B$ ) больше, чем  $B^J$  ( $B$  в степени  $J$ ), Джон выигрывает и отправляется на свидание, если  $B^J$  больше, чем  $J^B$  — то выигрывает Брюс. В случае равенства чисел процедура повторяется с самого начала.

Вычислите  $P_J$  — вероятность того, что выиграет Джон, и  $P_B$  — вероятность того, что выиграет Брюс.

### Формат входного файла

В первой строке входного файла содержится целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример состоит из одной строки, содержащей четыре целых числа  $J_1, J_2, B_1$  и  $B_2$  ( $1 \leq J_1 < J_2 \leq 10^9, 1 \leq B_1 < B_2 \leq 10^9$ ) в указанном порядке. Числа в строке разделены пробелами.

### Формат выходного файла

Для каждого тестового примера выведите в отдельной строке вероятности  $P_J$  и  $P_B$ , разделённые пробелом. Вероятности должны быть представлены в виде несократимых дробей в виде  $a/b$ , где  $a$  и  $b$  — целые,  $0 \leq a$  и  $b > 0$ .

### Пример

standard input	standard output
2	1/3 2/3
1 3 2 4	1/1 0/1
4 7 44 77	

## Задача В. The Cities

Имя входного файла:            standard input  
Имя выходного файла:           standard output  
Ограничение по времени:       5 seconds  
Ограничение по памяти:         256 Mebibytes

Джон и Брюс вспоминают о своих путешествиях по разным городам мира.

*John: Brus, have you ever been to Albania?*

*Brus: No, but I guess it should be a beautiful city...*

Джон и Брюс посетили  $N$  городов. Джон записал на листке бумаги названия этих городов. До тех пор, пока в списке остаётся не менее 3 городов, Брюс случайно выбирает 3 города из списка и ставит отметки на карте мира (карта является плоской). После чего Джон считает площадь треугольника, образованного этими городами, и вычёркивает эти 3 города из списка.

По окончании этого увлекательного процесса Брюс суммирует все площади, которые вычислил Джон. Ваша задача — найти математическое ожидание этой суммы. Обратите внимание, что некоторые треугольники могут быть и вырожденными...

### Формат входного файла

Первая строка входного файла содержит целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример начинается со строки, содержащей целое число  $N$  ( $1 \leq N \leq 1000$ ) — количество городов. Следующие  $N$  строк содержат координаты городов, которые посетили ребята —  $X_i$  и  $Y_i$  ( $-1000 \leq X_i, Y_i \leq 1000$ ). При этом все точки  $(X_i, Y_i)$  различны.

### Формат выходного файла

Для каждого тестового примера в отдельной строке выведите математическое ожидание суммы площадей. Ответы округляйте (согласно стандартных правил округления) до 7 знаков после запятой.

### Пример

standard input	standard output
2	58.8285714
7	0.0000000
-3 5	
-10 -8	
6 10	
8 6	
8 3	
-1 8	
4 4	
2	
4 7	
9 -1	

## Задача C. The Unfair Game

Имя входного файла:            standard input  
Имя выходного файла:           standard output  
Ограничение по времени:       2 seconds  
Ограничение по памяти:         256 Mebibytes

Джон и Брюс играют в следующую игру на доске  $H \times W$ .

Фишка Джона изначально расположена на поле  $(R_J, C_J)$ . Фишка Брюса изначально расположена на поле  $(R_B, C_B)$  (нумерация полей по вертикали и горизонтали начинается с единицы).

Игроки ходят по очереди. В свой ход Джон выбирает одно из четырёх направлений (вправо, влево, вверх, вниз) и двигает свою фишку на одну клетку в выбранном направлении. Брюс на своём ходу также может выбрать одно из этих четырёх направлений, и двигает свою фишку на любое целое неотрицательное количество клеток в выбранном направлении.

Игрок побеждает, если он на своём ходу может поставить фишку в клетку, занятую фишкой его оппонента.

*John: Brus, do you really think it's fair?*

*Brus: Sure, just go on!*

Определите результат игры, если оба игрока придерживаются оптимальной стратегии. Если игрок может выиграть, он будет стремиться минимизировать количество ходов в игре. Если игрок не может выиграть, он будет стремиться максимизировать количество ходов в игре.

### Формат входного файла

В первой строке содержится одно целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример начинается со строки, содержащей два разделённых пробелом целых числа  $H$  и  $W$ . Следующая строка содержит четыре разделённых пробелом целых числа  $R_J, C_J, R_B$  и  $C_B$  ( $1 \leq R_J, R_B \leq H$ ,  $1 \leq C_J, C_B \leq W$ ), в указанном порядке, при этом  $(R_J, C_J)$  и  $(R_B, C_B)$  различны. Последняя строка тестового примера содержит имя игрока, который начинает игру (John или Brus).

### Формат выходного файла

Для каждого тестового примера выведите в отдельной строке имя победителя и общее количество ходов в игре, разделённые пробелом.

### Пример

standard input	standard output
2	Brus 6
4 7	John 2
1 1 2 7	
John	
2 2	
1 2 2 1	
Brus	

## Задача D. The Almost Lucky Numbers

Имя входного файла:           standard input  
Имя выходного файла:         standard output  
Ограничение по времени:      2 seconds  
Ограничение по памяти:        256 Mebibytes

Согласно древней и почти забытой традиции 4-я задача должна быть посвящена «счастливым числам».

*John: Brus, Brus! I almost did it!*

*Brus: So, you are almost lucky!*

По некоторой версии, «счастливых» цифр всего две — 4 и 7. Джон и Брюс решили назвать цифры 3, 5, 6 и 8 «почти счастливыми», так как каждая из них отличается от какой-то «счастливой» цифры на единицу.

Назовём «почти счастливым» целое положительное число, если оно содержит в десятичной записи только «счастливые» или «почти счастливые» цифры и количество «счастливых» цифр не менее количества «почти счастливых».

Например, 7, 48 и 444773 — «почти счастливые» числа, а 485, 404 and 556 — нет.

Ваша задача — найти количество «почти счастливых» чисел между  $A$  и  $B$  включительно.

### Формат входного файла

Первая строка содержит целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример состоит из одной строки, содержащей 2 целых числа  $A$  и  $B$  ( $1 \leq A \leq B \leq 10^{18}$ ), разделённых пробелом.

### Формат выходного файла

Для каждого тестового примера выведите на отдельной строке число «почти счастливых» чисел между  $A$  и  $B$  включительно.

### Note

Например, между 10 и 100 «почти счастливыми» являются 34, 37, 43, 44, 45, 46, 47, 48, 54, 57, 64, 67, 73, 74, 75, 76, 77, 78, 84 и 87.

### Пример

standard input	standard output
2	2
1 10	20
10 100	

## Задача E. The Flights (1st Div Only)

Имя входного файла:            standard input  
Имя выходного файла:           standard output  
Ограничение по времени:       20 seconds  
Ограничение по памяти:         256 Mebibytes

На распродаже по случаю кризиса Джон и Брюс на карманные деньги купили несколько небольших самолётов. Они собираются открыть свою авиакомпанию и сделать бизнес на перевозках.

Всего есть  $N$  городов с аэропортами, при этом возможна организация авиарейса между любыми двумя из них. Брюс вычислил, что наиболее эффективно при имеющихся ресурсах будет организовать  $N$  авиарейсов так, что для каждого города есть ровно один рейс, прибывающий в город, и ровно один рейс, отправляющийся из этого города.

*John: What is this button for?*

*Brus: Damn, I've forgotten my parachute!*

Однако некоторые города соединены двунаправленными дорогами, при этом между двумя городами может быть более одной дороги. Если из одного города можно проехать в другой по дорогам (возможно, с заездом в другие города), то такие два города мы назовём связанными.

Джон предполагает, что организация авиарейса между двумя связанными городами экономически невыгодна. Поэтому он предлагает распределить все  $N$  маршрутов так, чтобы не было ни одного рейса между двумя связанными городами.

Ваша задача — подсчитать количество различных схем авиарейсов, которые удовлетворяют этому предложению, по модулю 1234567891.

### Формат входного файла

В первой строке содержится целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример начинается со строки, содержащей 2 целых числа  $N$  и  $M$  — количество городов и дорог соответственно. ( $1 \leq N \leq 1000$ ,  $1 \leq M \leq 10^5$ ). Каждая из последующих  $M$  строк содержит два разделённых пробелом различных числа  $A_i$  и  $B_i$  — города, соединённые  $i$ -ой дорогой ( $1 \leq A_i, B_i \leq N$ ).

### Формат выходного файла

Для каждого тестового примера выведите в отдельной строке количество различных схем авиарейсов, удовлетворяющих высказанному предложению, взятое по модулю 1234567891.

### Пример

standard input	standard output
2	4
4 3	1334961
1 2	
3 4	
4 3	
10 0	

## Задача F. The Easiest Problem

Имя входного файла:            standard input  
Имя выходного файла:           standard output  
Ограничение по времени:       2 seconds  
Ограничение по памяти:         256 Mebibytes

Согласно передаваемым из поколения в поколение традициям, в каждом констесте должна быть хотя бы одна простая задача, причём она должна быть спрятана где-то в середине. Если Вы искали её здесь — Вы угадали.

Джон и Брюс готовятся к финалу Чемпионата Мира по программированию. В процессе подготовки они решили какое-то количество задач на сервере asm.lviv.ua.

*John: Hey, Brus, have you solved this problem?*

*Brus: John, leave me alone — I'm sleeping!*

Вам задан список задач, решённых соответственно Джоном и Брюсом. Вам необходимо подсчитать 2 числа:  $X$  — количество различных задач, решённых хотя бы одним из ребят и  $Y$  — количество задач, которые решили и тот, и другой.

### Формат входного файла

В первой строке входного файла задано целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример начинается со строки, содержащей два целых числа  $N_J$  и  $N_B$  — количество задач, которые сдал Джон и количество задач, которые сдал Брюс, соответственно ( $1 \leq N_J, N_B \leq 100$ ).

Следующая строка содержит  $N_J$  целых чисел  $J_i$ , где  $J_i$  — номер  $i$ -й задачи, которую сдал Джон. Следующая строка содержит  $N_B$  целых чисел  $B_i$ , где  $B_i$  — номер  $i$ -й задачи, которую сдал Брюс ( $1 \leq J_i, B_i \leq 10^4$ ).

### Формат выходного файла

Для каждого тестового примера выведите одну строку, содержащую два целых числа  $X$  (количество задач, решённых хотя бы одним) и  $Y$  (количество задач, решённых обоими), разделённые пробелом.

### Пример

standard input	standard output
2	5 1
5 3	4 3
1001 1002 1001 1006 1099	
1007 1007 1099	
4 7	
1 2 3 4	
1 1 1 4 2 2 2	

## Задача G. The Squares

Имя входного файла:            standard input  
Имя выходного файла:           standard output  
Ограничение по времени:       20 seconds  
Ограничение по памяти:         256 Mebibytes

Джон и Брюс учатся перемножать целые числа. На доске записаны  $N$  различных целых чисел, и каждый раз ребята выбирают некоторое подмножество и считают произведение всех чисел этого подмножества.

*John: And what if we'll choose these numbers?*

*Brus: And what if we'll not? I'm tired!*

При выборе некоторых подмножеств в результате умножения получается полный квадрат. Ваша задача — определить количество непустых подмножеств, обладающих подобным свойством.

### Формат входного файла

В первой строке задано целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример начинается со строки, содержащей целое число  $N$  ( $1 \leq N \leq 1000$ ) — количество выписанных на доске чисел. Следующая строка содержит  $N$  целых чисел  $A_i$  ( $1 \leq A_i \leq 1000$ , все  $A_i$  попарно различны), разделённых пробелами.

### Формат выходного файла

Для каждого тестового примера выведите в отдельной строке количество непустых подмножеств, при перемножении элементов которых получается полный квадрат.

### Пример

standard input	standard output
2	3
4	7
3 6 2 1	
3	
4 9 16	

## Задача H. The Fast Reading

Имя входного файла:           standard input  
Имя выходного файла:         standard output  
Ограничение по времени:       10 seconds  
Ограничение по памяти:         256 Mebibytes

Брус пытается выработать навык быстрого чтения. Для этого он читает и произносит вслух различные строки, записанные строчными латинскими буквами ('a' — 'z'). Чтение и произношение любой буквы, кроме 'r', занимает у него одну секунду. Время на чтение и произношение 'r' зависит от буквы, которая идёт перед ней.

*John: Try this «rarorureriry».*

*Brus: Ha-ha, very funny!*

Джон собирается подготовить несколько строк, чтобы Brus смог на них тренироваться. У них не так много времени, поэтому John выбирает только непустые строки, которые Brus сможет прочитать не более, чем за  $N$  секунд. Ваша задача - подсчитать количество таких строк по модулю 1234567891.

### Формат входного файла

Первая строка входного файла содержит целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример начинается со строки, содержащей целое число  $N$  ( $1 \leq N \leq 10^9$ ). В следующей строке содержатся 27 целых чисел  $A_i$ , разделённых пробелами.  $A_1$  — время, за которое Brus читает и произносит букву 'r', когда она появляется в начале строки,  $A_2$  — когда 'r' следует за 'a',  $A_3$  — за 'b' и так далее. При этом ( $1 \leq A_i \leq 10$ ).

### Формат выходного файла

Для каждого тестового примера в отдельной строке выведите количество удовлетворяющих условию строк по модулю 1234567891.

### Пример

standard input	
2	
1	
2	1 1
4	
3	1 1 9 1 3 1 1 1 7 5 8 7 1 7 9 7 9 7 7 10 6 3 1 3 9 6
standard output	
25	
422401	



## Задача I. The Best Picture

Имя входного файла:            standard input  
Имя выходного файла:           standard output  
Ограничение по времени:       8 seconds  
Ограничение по памяти:         256 Mebibytes

Зимой Джон и Брюс играют на улице в кёрлинг. Так как никто больше не знает правил этой игры, то ребята играют один на один. При этом игры собирают большое количество зрителей, и вот уже поклонницы того или другого игрока создают фан-клубы... Особую активность проявил фан-клуб Джона.

*John: I'm not sure whether she is looking at me or you!*

*Brus: But I'm sure you are staring at her now.*

Однажды ребята объявили соревнование по следующим правилам. Они играют одну партию в каждый из  $N$  последующих дней. Если Джон выигрывает партию, он получает медаль от своего фан-клуба, если нет — он возвращает все медали в фан-клуб, (перед началом соревнования все медали находятся в фан-клубе). После каждой игры Джон, Брюс и поклонницы фотографируются на память. При этом Джон надевает все медали, которые у него есть. После того, как соревнование закончится, Джон выберет лучшую из  $N$  фотографий (ту, на которой у него наибольшее количество медалей) и разместит её на своей web-странице.

Вероятность, что Джон выиграет какую-то партию, равна  $P$ . Ваша задача — подсчитать математическое ожидание количества медалей на фотографии, которая будет размещена на web-странице.

### Формат входного файла

В первой строке содержится целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример состоит из одной строки, содержащей целое число  $N$  — количество дней, а также заданную ровно с четырьмя знаками после запятой вероятность  $P$  ( $1 \leq N \leq 1000$ ), ( $0 \leq P \leq 1$ ).

### Формат выходного файла

Для каждого тестового примера выведите строчку, содержащую математическое ожидание количества медалей у Джона на размещённой фотографии. Ответы округлять (согласно стандартных правил округления) до 7 знаков после запятой.

### Пример

	standard input	standard output
2		1.3750000
3	0.5000	0.4774000
1	0.4774	

## Задача J. The Flowers

Имя входного файла:	standard input
Имя выходного файла:	standard output
Ограничение по времени:	16 seconds
Ограничение по памяти:	256 Mebibytes

Джон и Брюс часто бывают в лесу недалеко от их дома. В лесу есть  $N$  полян и  $N - 1$  тропинок между ними (по каждой тропинке можно двигаться в обе стороны). Известно, что между любыми двумя полянами можно пройти по тропинкам, при этом между ними существует единственный простой путь (путь, не проходящий два раза через одну и ту же поляну).

*John: Brus, these flowers are for you!*

*Brus: Nice, they are beautiful! But why there are four of them?*

Ребята ходят в лес  $M$  дней подряд. Перед тем, как они в первый раз появились в лесу, на  $i$ -й поляне росло  $F_i$  цветов. В начале каждого дня на каждой поляне появляется один цветок. После этого John и Brus приходят в лес. За день они могут или сорвать все цветы на некоторой поляне, или подсчитать количество цветов на простом пути между двумя полянами (включая начальную и конечную)

Ваша задача — написать программу, которая бы дублировала каждый подобный подсчёт. inclusive.

### Формат входного файла

В первой строке входного файла содержится целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестов. Каждый тест начинается со строки, содержащей целые числа  $N$  и  $M$  ( $1 \leq N, M \leq 10^5$ ) — количество полян в лесу и количество дней, в течение которых ребята ходили в лес.

Следующая строка содержит  $N$  целых чисел  $F_i$  ( $0 \leq F_i \leq 10^9$ ) — количество цветов на  $i$ -й поляне.

Каждая из следующих  $N - 1$  строк содержит целые числа  $A_i$  и  $B_i$  — поляны, соединённые  $i$ -й тропинкой. Каждая из следующих  $M$  строк начинается с одного символа ('P' или 'C'). Если задан символ 'P', то за ним будет следовать целое число  $G_i$ , и это значит, что на  $i$ -й день ребята срывают все цветы с поляны  $G_i$ . Если задан символ 'C', то после него идут целые числа  $U_i$  и  $V_i$ . В этом случае Джон и Брюс подсчитывают количество цветов на простом пути между полянами  $U_i$  и  $V_i$ , включая начальную и конечную поляны. ( $1 \leq A_i, B_i, G_i, U_i, V_i \leq N$ ).

### Формат выходного файла

Для каждой команды 'C' в каждом тестовом примере выведите в отдельной строке искомое количество цветов.

## Пример

standard input	standard output
2	13
2 3	8
4 7	101
1 2	
C 1 2	
P 2	
C 1 2	
1 1	
100	
C 1 1	

## Задача К. The Card Game

Имя входного файла:           standard input  
Имя выходного файла:       standard output  
Ограничение по времени:     2 seconds  
Ограничение по памяти:       256 Mebibytes

Вечером Джон и Брюс играют в забавную карточную игру. В игре участвуют 16 попарно различных карт, каждая из которых может быть одной из четырёх мастей — Spades (пики), Clubs (трефы), Diamonds (бубны) или Hearts (черви), и с одной из четырёх картинок — валет, дама, король или туз (картинки перечислены в порядке возрастания старшинства карт)

Обозначим масть и картинку одной заглавной латинской буквой — ‘S’ (пики), ‘C’ (трефы), ‘D’ (бубны), ‘H’ (черви), ‘J’ (валет), ‘Q’ (дама), ‘K’ (король) or ‘A’ (туз). Тогда каждая карта может быть представлена как строка из двух символов, где первый символ обозначает картинку, а второй — масть карты.

*John: Can't find any joker!*

*Brus: Maybe, it's because I got them all!*

В начале игры карты перемешиваются и каждый из двух игроков получает по 8 карт.

Игроки ходят по очереди. На каждом ходу игрок или кладёт на стол навверх открытой колоды (изначально она пуста) одну из карт, старшинство которой не меньше, чем текущая верхняя карта открытой колоды (или любую карту, если открытая колода пуста), или берёт 3 карты из открытой колоды (если открытая колода содержит менее, чем 3 карты, игрок берёт все оставшиеся карты). Если после хода у какого-то игрока не осталось карт, он объявляется победителем.

Определите результат игры, если оба игрока придерживаются оптимальной стратегии. Если игрок может выиграть, он будет стремиться минимизировать количество ходов в игре. Если игрок не может выиграть, он будет стремиться максимизировать количество ходов в игре.

### Формат входного файла

В первой строке содержится одно целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример состоит из одной строки, содержащей 8 попарно различных карт  $C_i$ , которые John получил в начале игры. Карты описаны в соответствии с условием задачи и их описания разделены пробелами. Следующая строка содержит имя игрока, делающего первый ход (John или Brus).

### Формат выходного файла

Для каждого тестового примера в отдельной строке выведите через пробел имя игрока и общее количество в игре, или Draw, если игра будет длиться бесконечно долго.

### Пример

standard input	standard output
2	John 16
AS AC AD AH KS KC KD KH	Draw
Brus	
JS JC QD QH KS KC AD AH	
John	

## Задача L. The Balloons (2nd Div Only)

Имя входного файла:           standard input  
Имя выходного файла:         standard output  
Ограничение по времени:       2 seconds  
Ограничение по памяти:        256 Mebibytes

После завершения финала Чемпионата Мира по программированию у Джона осталось  $N$  воздушных шариков, а у Брюса —  $M$ .

*John: What is your favorite color?*

*Brus: The first one, of course!*

Ребята пытаются достичь соотношения количества шариков  $A : B$ .

Если  $N \times B$  больше, чем  $M \times A$ , Джон прокалывает один из своих воздушных шариков, а если  $N \times B$  меньше, чем  $M \times A$ , Брюс прокалывает один из своих.

Процесс продолжается до тех пор, пока не будет достигнуто соотношение количества шариков  $A : B$  или пока не будут проколоты все имеющиеся у ребят шарiki.

Ваша задача — подсчитать, сколько воздушных шариков будут иметь Джон и Брюс по окончании этого процесса.

### Формат входного файла

В первой строке содержится целое число  $T$  ( $1 \leq T \leq 47$ ) — количество тестовых примеров. Каждый тестовый пример состоит из одной строки, содержащей 4 целых числа  $N$ ,  $M$ ,  $A$  и  $B$  ( $1 \leq N, M, A, B \leq 10^9$ ), заданных в указанном порядке.

### Формат выходного файла

Для каждого тестового примера выведите в отдельной строке, сколько воздушных шариков по окончании процесса будет иметь Джон и сколько — Брюс.

### Пример

standard input	standard output
2	4 6
5 8 2 3	0 0
1 4 10 35	