

Задача А. Маленькие числа

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

У мальчика Влада есть два любимых числа a и b . Недавно его в школе научили делить и умножать, и он сразу побежал делить и умножать свои любимые числа.

Сначала он написал в тетрадке числа a и b , после чего решил, что будет последовательно производить с ними одну из трёх операций:

- Поделить оба числа на какой-то из их общих делителей;
- Поделить a на один из его делителей g , а b умножить на g ;
- Поделить b на один из его делителей g , а a умножить на g .

После каждой операции он стирает старые числа, записывает два получившихся числа обратно в тетрадку и может продолжить операции с ними.

Так как Влад ещё маленький, ему нравятся числа поменьше, потому он стремится минимизировать сумму чисел, написанных в тетрадке. Сам он не справляется. Помогите Владу определить, какую минимальную сумму чисел возможно получить такими операциями, и приведите пример пары чисел, которая может получиться в итоге.

Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t ($1 \leq t \leq 500$).

Каждый тест описывается следующим образом: в единственной строке описания теста содержатся два числа a и b ($1 \leq a, b \leq 10^9$) — любимые числа Влада.

Формат выходных данных

Для каждого теста в отдельной строке выведите ответ на него — пару с минимальной суммой, которую можно получить применяя операции из условия.

Если ответов несколько, то разрешается вывести любой из них.

Пример

стандартный ввод	стандартный вывод
2	1 5
4 5	2 3
4 6	

Задача В. Новая клавиатура

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Петя купил новую клавиатуру. Она поддерживает n раскладок. В каждой раскладке можно набирать некоторое подмножество строчных букв латинского алфавита. Пронумеруем раскладки от 1 до n .

Петя хочет набрать некоторое сообщение, состоящее из m строчных букв латинского алфавита. Исходно активна первая раскладка. Петя может производить следующие действия:

- Переключить раскладку. Тогда, если текущая раскладка имела номер i , новая раскладка будет иметь номер $i \bmod n + 1$, где $\bmod$ — операция взятия остатка по модулю. Если предыдущим действием Петя также переключал раскладку, это действие займет b миллисекунд, иначе это действие займет a миллисекунд.
- Набрать символ. Петя может добавить в конец текущего сообщения любую букву, содержащуюся в текущей раскладке. На это действие он потратит c миллисекунд.

Помогите Пете определить минимальное время, необходимое, чтобы набрать сообщение, либо выясните, что набрать сообщение невозможно. Раскладка, которая останется включенной после набора сообщения, может быть любой.

Формат входных данных

В первой строке содержатся четыре целых числа n , a , b и c — количество раскладок у клавиатуры, и количество миллисекунд, необходимых для совершения переключения раскладки и набора символа ($1 \leq n \leq 2000$, $1 \leq b \leq a \leq 10^9$, $1 \leq c \leq 10^9$).

В следующих n строках содержится описание раскладок. Каждая раскладка описывается непустой строкой, в которой каждая строчная буква латинского алфавита встречается не более одного раза — подмножество букв, которые можно набрать в этой раскладке. Буквы в этой строке упорядочены в алфавитном порядке.

В последней строке содержится строка s — сообщение, которое хочет набрать Петя (длина строки s от 1 до 2000). Строка s состоит из строчных латинских букв.

Формат выходных данных

Выведите единственное целое число — минимальное количество миллисекунд, необходимое, чтобы набрать сообщение. Выведите -1 , если набрать сообщение невозможно.

Примеры

стандартный ввод	стандартный вывод
5 3 2 1 abc d e f def abcdef	15
1 1 1 1 a z	-1

Задача С. Складывание фигуры

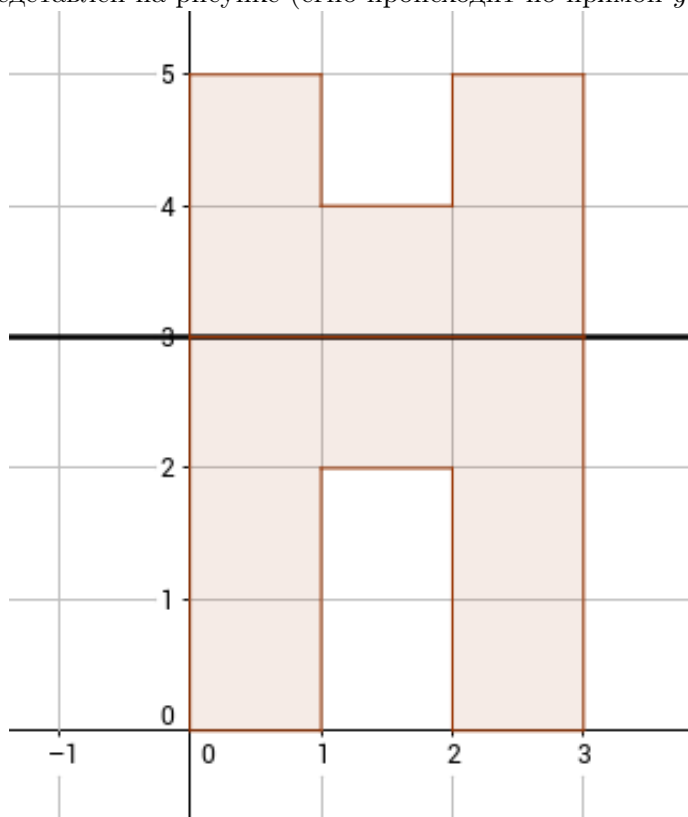
Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Пете как всегда стало скучно на уроке математики, поэтому он стал закрашивать в своей тетради клетки. Когда ему это надоело, он заметил, что у него получилась связная клетчатая фигура из k клеток — между любыми двумя закрашенными клетками существует путь по другим закрашенным клеткам, причем соседние клетки в этом пути имеют общую сторону.

Аккуратно вырезав ее из тетради, он сложил ее пополам вдоль линии клетчатой сетки (по горизонтали или по вертикали — он точно не запомнил), а также нарисовал копию сложенной фигуры в тетради. И не зря! Петя потерял фигуру, и все что у него осталось — копия сложенной фигуры, нарисованная в тетради. Теперь он хочет восстановить исходную фигуру.

Восстановить исходную фигуру однозначно не всегда возможно, однако Петя решил, что ему достаточно будет нарисовать хотя бы какую-нибудь фигуру из k клеток, которую можно сложить пополам так, чтобы получилась имеющаяся у него сложенная фигура. Помогите ему — найдите любую исходную связную фигуру из k клеток, удовлетворяющую этому условию.

Рассмотрим второй пример теста из условия. В нем сложенная фигура представляет собой букву «П», а исходная фигура состоит из 12 клеток. Один из возможных вариантов исходной фигуры представлен на рисунке (сгиб происходит по прямой $y = 3$):



Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t ($1 \leq t \leq 200$).

Каждый из следующих t тестов описывается следующим образом: в первой строке описания теста содержится два целых числа n , k — количество закрашенных клеток, из которых состоит сложенная фигура Васи и количество клеток в исходной фигуре ($1 \leq n < k \leq 10^5$).

В каждой из следующих n строк содержатся два числа x_i , y_i — координаты левого нижнего угла i -й закрашенной клетки ($-10^8 \leq x_i, y_i \leq 10^8$). Гарантируется, что все закрашенные клетки различны и образуют связную фигуру.

Гарантируется, что сумма k во всех тестах одних входных данных не превосходит 10^5 .

Формат выходных данных

Для каждого теста выведите ответ на него. Необходимо вывести описание фигуры и способа согнуть ее, чтобы получить фигуру из входного файла.

В первой строке выведите линию сгиба, а в k следующих строках по два целых числа (x'_i, y'_i) — координаты клеток связной фигуры, которую можно сложить пополам по выведенной линии сгиба, чтобы получить фигуру во входных данных.

Линия сгиба должна быть выведена одним из 4 способов:

- L num — сгиб производится по прямой $x = num$, левая часть накладывается поверх правой;
- R num — сгиб производится по прямой $x = num$, правая часть накладывается поверх левой;
- U num — сгиб производится по прямой $y = num$, верхняя часть накладывается поверх нижней;
- D num — сгиб производится по прямой $y = num$, нижняя часть накладывается поверх верхней.

Все x'_i, y'_i , а также координата линии сгиба по модулю не должны превышать 10^9 . Гарантируется, что такая фигура существует. Если существует несколько подходящих ответов, разрешается вывести любой из них.

Пример

стандартный ввод	стандартный вывод
2	L 0
7 14	0 0
0 0	0 1
0 1	0 2
0 2	1 2
1 2	2 0
2 2	2 1
2 1	2 2
2 0	-1 0
7 12	-1 1
0 0	-1 2
0 1	-2 2
0 2	-3 2
1 2	-3 1
2 2	-3 0
2 1	U 3
2 0	0 0
	0 1
	0 2
	1 2
	2 2
	2 1
	2 0
	0 3
	1 3
	2 3
	0 4
	2 4

Задача D. Остроугольные треугольники

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Совсем недавно московский десятиклассник Дмитрий Захаров обогнал всех в построении множества точек в d -мерном пространстве, что все треугольники с вершинами в этих точках остроугольные.

Таня решила померяться силами с Дмитрием. Разумеется, она планирует использовать компьютер. Для начала она хочет решить следующую задачу. Дано n точек на плоскости, требуется количество остроугольных треугольников с вершинами в этих точках. Треугольник называется остроугольным, если все его углы строго меньше 90 градусов.

Формат входных данных

Входные данные содержат несколько тестовых наборов. В первой строке задано количество тестов t ($1 \leq t \leq 666$).

Каждый из тестов описывается следующим образом: в первой строке описания теста содержится число n ($3 \leq n \leq 2000$) — количество точек.

В следующих n строках содержатся по два числа x_i, y_i ($-10^9 \leq x, y \leq 10^9$) — координаты точек. Гарантируется, что все точки в одном тесте различные.

Суммарное количество точек по всем тестам одних входных данных не превосходит 2000.

Формат выходных данных

Для каждого теста в отдельной строке выведите ответ на него — количество остроугольных треугольников с вершинами в заданных точках.

Пример

стандартный ввод	стандартный вывод
2	3
5	4
1 1	
2 2	
3 3	
4 1	
6 4	
5	
0 0	
3 1	
5 1	
5 -1	
1 3	

Задача Е. Объединение массивов

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Рассмотрим два массива натуральных чисел $A = [a_1, a_2, \dots, a_n]$ и $B = [b_1, b_2, \dots, b_m]$. Будем называть их k -объединением лексикографически минимальный массив чисел R длиной k , который разбивается на две непустые подпоследовательности, первая из которых является подпоследовательностью массива A , а вторая — подпоследовательностью массива B .

Формально говоря, необходимо найти лексикографически минимальный массив $R = [r_1, r_2, \dots, r_k]$, для которого существуют непустые наборы индексов $1 \leq i_{1,1} < i_{1,2} < \dots < i_{1,t} \leq n$ и $1 < j_{1,1} < j_{1,2} < \dots < j_{1,k-t} \leq m$ в исходных массивах, а также наборы индексов $1 \leq i_{2,1} < i_{2,2} < \dots < i_{2,t} \leq k$ и $1 \leq j_{2,1} < j_{2,2} < \dots < j_{2,k-t} \leq k$, такие что:

- Для любых $1 \leq p \leq t$, $1 \leq q \leq k - t$ выполнено $i_{2,p} \neq j_{2,q}$;
- Для любого $1 \leq p \leq t$ выполнено $a_{i_{1,p}} = r_{i_{1,p}}$;
- Для любого $1 \leq p \leq k - t$ выполнено $b_{j_{1,p}} = r_{j_{1,p}}$.

Например, если $A = [1, 2, 1, 3, 1, 2, 1]$, $B = [1, 2, 3, 1]$, а $k = 9$, то их k -объединением будет $R = [1, 1, 1, 1, 2, 1, 2, 3, 1]$ (подпоследовательность из первого массива — $[1, 1, 1, 2, 1]$, подпоследовательность из второго массива — $[1, 2, 3, 1]$).

По двум данным массивам A и B , а также числу k найдите их k -объединение R .

Формат входных данных

В первой строке ввода содержится число n — размер массива A ($1 \leq n \leq 3000$).

Во второй строке содержится n чисел a_i — массив A ($1 \leq a_i \leq 3000$).

В третьей строке содержится число m — размер массива B ($1 \leq m \leq 3000$).

Во четвертой строке содержится m чисел b_i — массив B ($1 \leq b_i \leq 3000$).

В последней строке содержится число k ($2 \leq k \leq n + m$).

Формат выходных данных

Выведите k -объединение заданных во входном файле массивов.

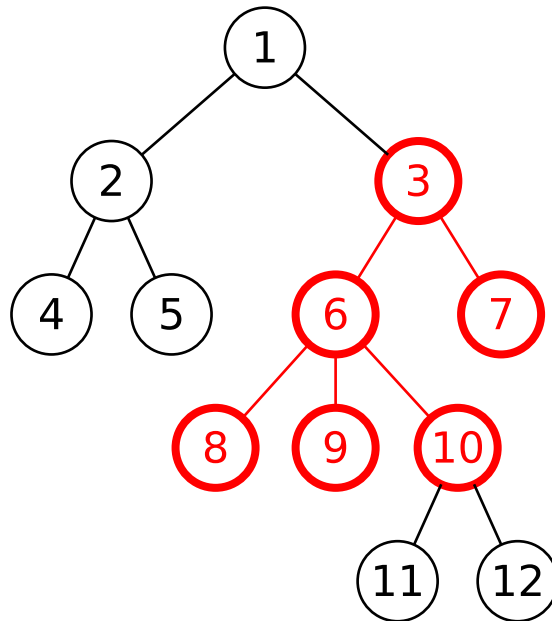
Пример

стандартный ввод	стандартный вывод
7 1 2 1 3 1 2 1 4 1 2 3 1 9	1 1 1 1 2 1 2 3 1

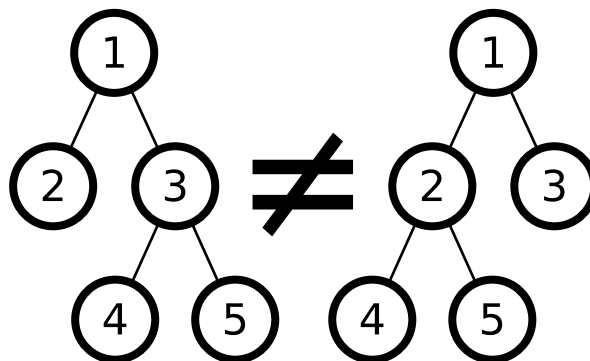
Задача F. Два поддерева

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 256 мегабайт

Рассмотрим подвешенное дерево. рассмотрим вершину v , имеющую в поддереве хотя бы одну вершину на расстоянии k от v . Будем называть k -поддеревом вершины v дерево, полученное из **поддерева** вершины v удалением всех вершин, расстояние от которых до v больше k . Например, на иллюстрации ниже красным отмечено 2-поддерево вершины 3.

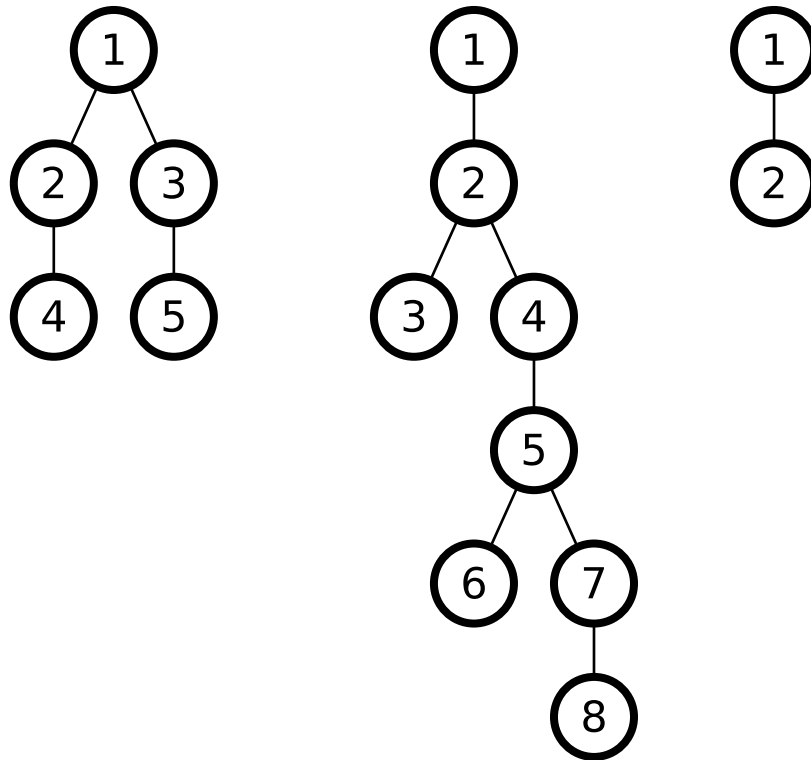


Будем называть два дерева одинаковыми, если можно перенумеровать вершины первого так, чтобы получить второе, при этом менять порядок детей у вершин не разрешается. Например, следующие два дерева **не являются** одинаковыми:



Дано подвешенное дерево. Требуется определить наибольшее k такое, что у него есть два одинаковых k -поддерева, корни которых различны. Эти поддерева могут пересекаться.

На рисунке приведены деревья из примеров.



В первом примере корни одинаковых 1-поддеревьев — это вершины 2 и 3.

Во втором примере корни одинаковых 3-поддеревьев — это вершины 1 и 4.

В третьем примере корни одинаковых 0-поддеревьев — это вершины 1 и 2.

Формат входных данных

В первой строке задано число t — количество тестов ($1 \leq t \leq 10^4$).

Каждый из t тестов описывается следующим образом: в первой строке задано число n — количество вершин ($2 \leq n \leq 10^5$).

Далее следует n строк. В i -й из них задано число cnt_i — количество детей вершины i , а далее следуют cnt_i чисел — номера детей вершины i . Вершины нумеруются с единицы. Корнем дерева является вершина 1. Гарантируется, что заданный граф является деревом с корнем в 1.

Сумма n по всем тестам в одних входных данных не превосходит $2 \cdot 10^5$.

Формат выходных данных

Для каждого теста выведите на отдельной строке максимальное k такое, что существует два одинаковых k -поддерева с разными корнями.

Пример

стандартный ввод	стандартный вывод
3	1
5	3
2 2 3	0
1 4	
1 5	
0	
0	
8	
1 2	
2 3 4	
0	
1 5	
2 6 7	
0	
1 8	
0	
2	
1 2	
0	