

Задача А. А-голов В-ног

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

2924 год. Человечество уже 6-й век живет на Марсе. Есть люди которые жили в хороших условиях и сохранили свою человеческую идентичность, у них 1 голова и 2 ноги. А есть и те, которые страдали от солнечного излучения и еще много чего, у них A -голов и B -ног, назовем их АБшки.

Даны 2 числа, n и m – количество голов и ног у всех жителей Марса. Помогите понять, сколько из них люди, а сколько АБшки.

Формат входных данных

Даны 4 числа – A, B, n и m ($1 \leq A, B \leq 10^9, 0 \leq n, m \leq 10^9$).

Формат выходных данных

Выведите 2 числа: возможное количество людей и АБшек. Если решений несколько, выведите то, в котором больше всего людей и меньше всего АБшек. Если решения нет, выведите -1 .

Система оценки

Задача состоит из трёх подзадач.

- Подзадача 1(20 баллов): $B \neq 2 \cdot A$, гарантируется, что решение существует.
- Подзадача 2(30 баллов): гарантируется, что решение существует, необходимые подзадачи – 1
- Подзадача 3(50 баллов): без дополнительных ограничений, необходимые подзадачи – 1, 2

Примеры

стандартный ввод	стандартный вывод
1 4 3 8	2 1
2 4 1 2	1 0
1 3 1 4	-1

Задача В. Циферблат

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Есть n циферблатов, пронумерованных от 1 до n , на каждом из которых написано целое число, при этом на i -м записано число a_i . Это число можно увеличить на b_i , потратив на это c_i сил, или уменьшить на b_i за d_i сил. Нужно, потратив как можно меньше сил, сделать произведение чисел на всех циферблатах строго положительным.

Формат входных данных

В первой строке входного файла дано число $1 \leq n \leq 10^5$. В следующих четырех строках даны по n чисел – массивы a, b, c, d соответственно. Выполнено ограничение: $|a_i| \leq 10^9, 1 \leq b_i, c_i, d_i \leq 10^9$.

Формат выходных данных

Выведите единственное число – минимальное количество сил, которое нужно потратить, чтобы сделать произведение положительным.

Примеры

стандартный ввод	стандартный вывод
5 0 0 0 0 0 1 1 1 1 1 1 2 1 2 1 2 1 2 1 2	5
3 -5 8 -2 4 2 1 3 1 2 1 2 10	0

Замечание

- Подзадача 1 (20 баллов): $a_i \neq 0$
- Подзадача 2 (25 баллов): $c_i = d_i = 1$
- Подзадача 3 (25 баллов): $c_i = d_i$, необходимые подзадача – 2
- Подзадача 4 (30 баллов): Без дополнительных ограничений, необходимые подзадачи – 1, 2, 3

Задача С. Привлекательное разбиение

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

У вас есть множество последовательных натуральных чисел от 1 до n и число m . Назовем число x *привлекательным*, если $x = k^2$ для некоторого натурального числа k или $x = k^2 \cdot p$, где k, p – натуральные числа и p – простое. Ваша задача – разбить числа из заданного множества на *не более* чем m непустых наборов так, чтобы сумма чисел в каждом наборе была привлекательным числом.

Формат входных данных

В первой строке вам даны число $1 \leq n \leq 10^5$, означающее, что у вас есть числа $1, 2, \dots, n$, и число $20 \leq m \leq 10^5$.

Формат выходных данных

В первой строке выведите число q – количество наборов в вашем разбиении. Должно быть выполнено ограничение $1 \leq q \leq m$. В следующих строках выведите информацию о ваших наборах. Каждое описание набора должно начинаться с числа k_i – количества чисел в наборе. В следующей строке выводите k_i чисел – элементы, которые попали в этот набор. Элементы в наборе можно выводить в любом порядке. Суммарный размер всех наборов должен быть равен n , каждое число от 1 до n должно содержаться ровно в одном наборе.

Система оценки

Задача состоит из четырёх подзадач.

Система оценки

- Подзадача 1(20 баллов) $n \leq 32, m = 10^5$
- Подзадача 2(16 баллов) $n = k^2 - 1, m = 10^5$ для некоторого натурального k, n – чётное
- Подзадача 3(18 баллов) $m = 10^5$, необходимые подзадачи – 1, 2
- Подзадача 4(46 баллов) $m = 20$, необходимые подзадачи – 1, 2, 3, 4

Пример

стандартный ввод	стандартный вывод
16 20	7 2 6 3 1 9 1 13 4 14 15 10 11 3 2 4 5 3 12 7 8 2 1 16

Задача D. Крош и пары соседних элементов

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

У Кроша есть массив из n целых чисел. Крош хочет, чтобы пары соседних элементов в массиве имели сумму, большую нуля, и он называет пару соседних элементов *плохой*, если она имеет сумму, меньше либо равную нулю. За одну операцию Крош может взять любой элемент и увеличить его на 1. Для каждого k от 0 до $n - 1$ выведите минимальное количество операций, которое нужно сделать Крошу, чтобы в массиве было не более k плохих пар.

Формат входных данных

Вам дано число $2 \leq n \leq 500$, обозначающее количество элементов в массиве. В следующей строке вам даны n целых чисел $-10^6 \leq a_i \leq 10^6$ – элементы массива.

Формат выходных данных

Выведите n чисел – для каждого $0 \leq k < n$ минимальное количество операций, которое нужно сделать Крошу, чтобы в массиве было не более k плохих пар.

Система оценки

- Подзадача 1(10 баллов): $n \leq 20$
- Подзадача 2(20 баллов): $n \leq 50$, $-100 \leq a_i \leq 100$
- Подзадача 3(30 баллов): $n \leq 50$, необходимые подзадачи – 2
- Подзадача 4(40 баллов): Без дополнительных ограничений, необходимые подзадачи – 1, 2, 3

Пример

стандартный ввод	стандартный вывод
3 -2 -6 -1	9 8 0

Задача Е. Цветная строка

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

Вы получили строку из n символов, где каждый элемент покрашен в один из m цветов, а цвета – натуральные числа от 1 до m . Вы можете делать над строкой некоторые операции. А именно, вы можете выбрать цвет x и перевести его в цвет y . Цена такой замены равна $cost_{xy}$. Все символы одного цвета переводятся одновременно. Дополнительное ограничение: символы, которые уже были преобразованы в какой-то другой цвет, дальше не меняются (даже в том случае, если мы выполним операцию над новым цветом – изменятся только нетронутые ранее символы этого цвета). Например, если была последовательность символов с цветами 1 5 1 7, и 1 перевели в 5, а потом 5 в 7, то итоговая последовательность символов будет 5 7 5 7. Вы хотите выполнить операции таким образом, чтобы итоговая последовательность цветов символов не убывала. Найдите минимальную суммарную стоимость, которую необходимо заплатить, чтобы достичь цели.

Формат входных данных

В первой строке даны два числа $2 \leq n \leq 5 \cdot 10^5$ и $2 \leq m \leq 1000$. Во второй строке даны n чисел $1 \leq a_i \leq m$. Далее в m строках идет по m чисел в каждой $0 \leq cost_{ij} \leq 10^9$, $cost_{ii} = 0$

Формат выходных данных

Выведите единственное число – ответ на задачу.

Система оценки

Задача состоит из пяти подзадач.

- Подзадача 1(10 баллов): Все недиагональные значения в $cost$ равны между собой, все a_i различны
- Подзадача 2(15 баллов): Все недиагональные значения в $cost$ равны между собой, необходимые подзадачи – 1
- Подзадача 3(20 баллов): Все a_i различны, необходимые подзадачи – 1
- Подзадача 4(20 баллов): $n \leq 1000$
- Подзадача 5(35 баллов): Без дополнительных ограничений, необходимые подзадачи – 1, 2, 3, 4

Пример

стандартный ввод	стандартный вывод
7 3 3 1 1 1 1 1 2 0 2 3 3 0 1 1 4 0	1

Задача F. Все дороги ведут в Рим!

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	1024 мегабайта

Все дороги ведут в Рим! Завтра в столице будут проводиться крупные торги, куда приедут торговцы со всех городов Империи.

В Римской Империи есть n городов, связанных между собой дорогами. Разумеется, из любого города можно добраться в Рим, и сделать это можно единственным образом, если не посещать никакой город дважды. На рассвете из некоторых городов в сторону столицы выедут караваны с товаром, каждый из них имеет свой коэффициент прибыли.

Об этом узнали грабители и решили устроить идеальное ограбление. Для этого они хотят расположиться в одном из городов, чтобы грабить все проезжающие через него караваны. Но в Римской империи есть своя система безопасности: чем ближе город к столице, тем более влиятельны там силы правопорядка. Если через город, находящийся на расстоянии d от Рима, проезжает караван с коэффициентом доходности c , то грабители, находясь в этом городе, смогут получить $(d + 1) \cdot c$ прибыли с этого каравана.

Так как грабители не знают, сколько караванов и из каких городов выдвинутся на торги, им интересно узнать, какую максимальную прибыль они могут получить для q гипотетических запросов.

Формат входных данных

Первая строка содержит одно целое число n ($2 \leq n \leq 4 \cdot 10^5$) — количество городов в империи. Города пронумерованы целыми числами от 1 до n . Рим — город номер 1.

В следующей $n - 1$ строке содержится описание дорог в виде двух чисел u и v ($1 \leq u, v, \leq n$). Это означает, что между городами u и v есть дорога.

Далее содержатся два целых числа q и t . q ($1 \leq q \leq 4 \cdot 10^5$) — количество запросов. t ($1 \leq t \leq 2$) — тип описания запросов. Затем следует описание запросов.

В начале каждого описания запроса содержится строка с одним целым числом k ($1 \leq k < n$) — количеством караванов.

Далее дается описание всех караванов, состоящее из k строк. В зависимости от t описание каравана имеет следующий вид:

Если $t = 1$ то в i -й строке содержится 2 целых числа — v_i ($2 \leq v_i \leq n$) и c_i ($1 \leq c_i \leq 10^6$). i -й караван выдвигается из города v_i и имеет коэффициент доходности c_i . Гарантируется что внутри одного запроса все v_i различны.

Если $t = 2$ то в i -й строке содержится 2 целых числа — u_i ($0 \leq u_i < n - 1$) и c_i ($1 \leq c_i \leq 10^6$). Пусть $last_{ans}$ — ответ на предыдущий запрос. Для первого запроса $last_{ans} = 0$. Тогда i -й караван текущего запроса выдвигается из города $(u_i + last_{ans}) \bmod (n - 1) + 2$, где mod обозначает операцию взятия остатка от деления. c_i соответственно коэффициент доходности i -го каравана. Гарантируется что внутри одного запроса все u_i различны.

Гарантируется что сумма k по всем наборам не превосходит $4 \cdot 10^5$.

Система оценки

Обозначим за K_{sum} сумму k по всем наборам.

Группа тестов	Дополнительные ограничения			Баллы	Необходимые группы
	n	K_{sum}	t		
1	$n \leq 3000$	$k \leq 3000$	1	10	
2	$n \leq 10^5$	$k \leq 10^5$	1	35	1
3	$n \leq 10^5$	$k \leq 10^5$	2	25	1 — 2
4	—	—	2	30	1 — 3

Обратите внимание, что для прохождения любой группы тестов ваша программа **не** обязана выдавать верный ответ на примерах из условия.

Формат выходных данных

Выведите q целых чисел. i -е число — максимальное значение прибыли для i -го запроса.

Примеры

стандартный ввод	стандартный вывод
4 1 2 2 3 3 4 1 1 2 2 10 3 3	26
5 1 2 3 1 4 3 3 5 2 2 3 0 7 3 4 2 4 3 1 4 2 2 3 1	16 14

Задача G. Игра на числах

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Двое игроков играют в игру. У них есть набор из n целых неотрицательных чисел, которые записаны в двоичной системе счисления друг за другом, слева направо (лидирующие нули в числах допустимы). За один ход игрок может взять некоторое положительное число, и, если оно делится на 2, вычесть из него 1, а если не делится на 2, то либо вычесть из него 1, либо поделить на 2 с округлением вниз. Проигрывает тот, кто не может сделать ход. Вы решили проверить этих игроков и предложить им сыграть большое количество раз. Пусть l_1, l_2, \dots, l_n – длины этих чисел. Вы пронумеровали все цифры всех чисел слева направо таким образом, что цифры первого числа пронумерованы от 1 до l_1 , цифры второго числа пронумерованы от $l_1 + 1$ до $l_1 + l_2$ и так далее. Вы берете все позиции чисел от L до R и делаете флип в этих цифрах, то есть меняете нули на единицы, а единицы на нули. До всех изменений и после каждого изменения игроки играют в игру, и ваша задача – определить, кто выиграет при оптимальной игре обоих игроков. Обратите внимание, что изменения сохраняются для следующих запросов.

Формат входных данных

В первой строке вам дано количество чисел и количество запросов $1 \leq n \leq 2 \cdot 10^5$ и $1 \leq q \leq 2 \cdot 10^5$. В следующих n строках вам заданы числа. Каждое число представляет из себя последовательность из нулей и единиц, записанных подряд без пробелов. Суммарная длина s всех чисел не превосходит $2 \cdot 10^5$. В следующих q строках вам заданы запросы. Каждый запрос представляет собой два числа $1 \leq l \leq r \leq s$ – границы позиций изменяемых цифр.

Формат выходных данных

Выведите $q + 1$ строк – ответы до запросов и после применения каждого из запросов. Выведите «First» (без кавычек), если выигрывает первый игрок, и «Second» (без кавычек), если выигрывает второй игрок.

Система оценки

- Подзадача 1 (5 баллов): $n = 1, s \leq 18$
- Подзадача 2 (11 баллов): $n \leq 100, q \leq 100$, длина каждого числа не превосходит 18
- Подзадача 3 (7 баллов): $n = 1$ без дополнительных ограничений, необходимые подзадачи – 1
- Подзадача 4 (11 баллов): $n \leq 100, q \leq 100$ без дополнительных ограничений, необходимые подзадачи – 2
- Подзадача 5 (15 баллов): $n \leq 100$, необходимые подзадачи – 2, 4
- Подзадача 6 (19 баллов): Длина каждого числа не превосходит 18, необходимые подзадачи – 1, 2
- Подзадача 7 (32 балла): Без дополнительных ограничений, необходимые подзадачи – 1, 2, 3, 4, 5, 6

Примеры

стандартный ввод	стандартный вывод
1 5 010101010 9 9 8 8 7 8 5 5 1 9	Second First First First First Second
2 7 1001101 0110 11 11 2 3 7 8 7 7 8 11 6 8 3 6	First Second Second First First First Second Second