

Задача А. Подмножества наносят ответный удар

Имя входного файла: `combo.in`
Имя выходного файла: `combo.out`
Ограничение по времени: 7 секунд
Ограничение по памяти: 512 мегабайт

Имея множество из n различных элементов, легко найти количество его различных k -элементных подмножеств. Это хорошо известный биномиальный коэффициент $\binom{n}{k} = \frac{n!}{k!(n-k)!}$.

Однако, данная задача несколько сложнее. Вам дано мультимножество, состоящее из n (возможно, совпадающих) элементов, и вам требуется ответить на несколько вопросов вида “если бы мы добавили элемент x в мультимножество, то сколько бы в нем существовало различных k -элементных подмультимножеств?”. Обратите внимание, что после каждого запроса множество **не изменяется**.

Напомним, что k -элементное подмультимножество — это любые k элементов исходного мультимножества (разрешается брать одинаковые элементы). Два мультимножества считаются различными, если какой-либо элемент встречается в них разное число раз.

Так как ответы на запросы могут быть слишком большими, от вас требуется вывести их по модулю 1 051 721 729.

Формат входных данных

В первой строке входных данных содержится число n — размер мультимножества ($1 \leq n \leq 120\,000$). В следующей строке содержатся n чисел a_i : элементы мультимножества ($1 \leq a_i \leq n$). Обратите внимание, что элементы могут совпадать. В следующей строке содержится число q — количество запросов ($1 \leq q \leq 120\,000$). Каждый из следующих q строк описывает запрос, описание запроса состоит из двух чисел x и k ($1 \leq x \leq n, 1 \leq k \leq n + 1$).

Формат выходных данных

Для каждого запроса выведите одно число — искомое количество подмультимножеств, взятое по модулю 1 051 721 729.

Примеры

<code>combo.in</code>	<code>combo.out</code>
6	6
1 2 2 3 3 3	7
4	6
1 2	8
2 3	
3 4	
4 5	

Замечание

При выполнении первого запроса (если мы добавим элемент 1), исходное мультимножество будет выглядеть как 1 2 2 3 3 3 1, таким образом, оно будет содержать 6 различных подмультимножеств размера 2: (1 1), (1 2), (1 3), (2 2), (2 3) и (3 3).

Задача В. Делитель многочлена

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

Петя узнал, что количество способов выбрать три шара из x равно $x(x-1)(x-2)/6$. Это означает, что многочлен $x(x-1)(x-2)$ делится на 6 для любого целого x . Пете понравилось это свойство, и он решил изучить его для других многочленов такого вида. Пусть у вас есть многочлен $P(x) = \prod_{i=0}^{n-1} (x-i)^{d_i}$ для каких-то целых n и d_i . Найдите наибольшее число, на которое $P(x)$ делится при любом целом x .

Поскольку ответ может быть большим, вычислите его по модулю $10^9 + 7$.

Формат входных данных

Первая строка входного файла содержит число n ($1 \leq n \leq 10^4$). Вторая строка содержит n чисел d_i ($0 \leq d_i \leq 9$).

Формат выходных данных

Выведите одно число — наибольший делитель по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
3 1 1 1	6
5 0 0 0 0 0	1
4 2 0 1 4	16

Задача С. Дружелюбные хомячки

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

На плоскости живут n хомячков. Каждый в точке с целыми координатами. Хомячки дружат, если существует прямоугольник со сторонами, параллельными осям координат, содержащий этих двух хомячков и не содержащий никаких других.

Прямоугольник содержит хомячка, если точка, в которой он живет, лежит внутри прямоугольника или на его границе.

Сколько пар хомячков дружат?

Формат входных данных

На первой строке число n , $1 \leq n \leq 100\,000$.

Следующие n строк содержат по два целых числа — координаты точек, в которых живут хомячки.

Все точки различны, а координаты целые, по модулю не превосходят 10^9 .

Формат выходных данных

Выведите одно целое число — количество пар дружащих хомячков.

Примеры

стандартный ввод	стандартный вывод
5 0 0 0 2 2 0 2 2 1 1	8

Задача D. Ковер

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

Все кинозвезды Бомжландии собираются вместе на церемонию вручения Бузкара. Обычно для кинозвезд расстилается красная ковровая дорожка, но ее съела моль, поэтому нужно срочно разместить новый ковер.

Бомжландия состоит из n городов, связанных между собой $n - 1$ двусторонними дорогами. Будем считать, что ковер представляет собой клеточный прямоугольник $1\,000\,000 \times 20$ квадратных метров. Города должны быть размещены в центрах клеток, при этом никакие два города не могут быть размещены в одной клетке. После этого дороги должны быть нарисованы как отрезки, соединяющие центры соответствующих клеток.

Ковер должны быть красивым, поэтому никакие два отрезка не должны пересекаться в точках, отличных от концов отрезков. Какой-то странный математик уже доказал, что красивый ковер всегда возможно сплести, однако, он не предоставил конструктивного решения. Вам же надо просто сплести ковер.

Формат входных данных

В первой строке входных данных содержится число n ($1 \leq n \leq 100\,000$) — число городов в Бомжландии. Следующие $n - 1$ строк содержат описания дорог: каждое описание состоит из двух чисел u_i и v_i ($1 \leq u_i, v_i \leq n$, $u_i \neq v_i$) — номеров городов, которые соединены соответствующей дорогой.

Формат выходных данных

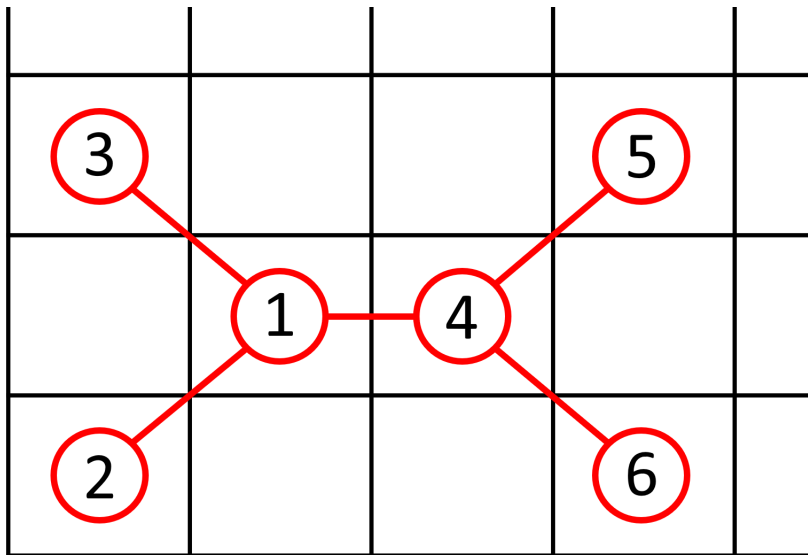
Выведите n строк: строка i должна содержать два целых числа x_i и y_i ($1 \leq x_i \leq 1\,000\,000$, $1 \leq y_i \leq 20$) — координаты клетки, в которой нужно разместить город i .

Примеры

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

Замечание

Картинка соответствует первому примеру:



Задача Е. Высокий маразм

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 5 секунд
Ограничение по памяти: 64 мегабайта

Город Нью-Васюки представляет собой таблицу N на N кварталов. Таким образом, в нем существует $(N + 1)(N + 1)$ перекрестков и $2N(N + 1)$ двусторонних дорог. Каждый перекресток имеет высоту. Известно, что левый верхний перекресток имеет высоту 0, а правый нижний – высоту 1. Для каждой дороги известно, сколько человек идет в каждом направлении по этой дороге. При этом, если дорога ведет от перекрестка i к перекрестку j , и разность высот $h = h_j - h_i$, то неудобство перемещения для каждого человека равно $\max(h, 0)$.

Для всех клеток, кроме двух угловых вы вправе выбирать любую высоту. Найдите распределение, при котором суммарное неудобство будет минимальным.

Формат входных данных

В первой строке входного файла содержится число N – размер таблицы, $1 \leq N \leq 500$. Далее следует $4N(N+1)$ чисел – искомое число людей для каждой дороги в естественном порядке. Сначала следует $N(N + 1)$ чисел для каждой дороги с запада на восток, затем столько же чисел с севера на юг, потом столько же чисел с востока на запад, а затем столько же чисел с юга на север. Для каждого направления дороги перечислены с севера на юг, а затем – с запада на восток. Обратите внимание на пример (там все числа специально различны).

Все количества являются целыми неотрицательными числами не превосходящими миллиона.

Формат выходных данных

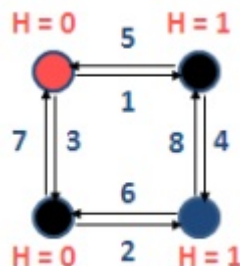
Выведите одно число – итоговое минимальное суммарное неудобство.

Примеры

stdin	stdout
1	3
1	
2	
3	
4	
5	
6	
7	
8	

Замечание

Высоты могут быть произвольными вещественными числами! Картинка к примеру из условия:



Задача F. Мухобойка

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

Вы знаете разницу между отелем и мотелем? Вы правы, разница в количестве мух, живущих там. Петя владелец одного из самых популярных мотелей в Берляндии, но его мама настаивает на том, чтобы он превратил свой мотель в отель. Именно поэтому она подарила Пете мухобойку в виде многоугольника из k вершин.

Подойдя к окну Петя увидел n мух. Так как Петя и мухи не обидит, он хочет узнать количество способов ударить мухобойкой, не покалечив ни одной мухи.

Окно представляет из себя прямоугольник, нижний левый угол которого находится в центре координатной системы. После удара Пети все вершины мухобойки должны находиться в целых координатах и не должны вылезать за границы окна. Муха покалечена, если она будет находиться под мухобойкой либо на её границах.

Формат входных данных

Первая строка входных данных содержит числа x_p , y_p и n ($1 \leq x_p, y_p \leq 500$, $0 \leq n \leq x_p \cdot y_p$) — координаты правого верхнего угла окна и количество мух на ней, соответственно.

Следующие n строк содержат по два числа x и y ($0 < x < x_p$, $0 < y < y_p$) — координаты мух на окне.

Следующая строка содержит число k ($3 \leq k \leq 10\,000$) — количество вершин у мухобойки. В следующих k строках даются x_i и y_i ($-10^9 \leq x_i, y_i \leq 10^9$) — координаты i -й вершины мухобойки. Вершины заданы в порядке обхода, так что соседние вершины и первая и последняя вершины соединены прямой линией.

Формат выходных данных

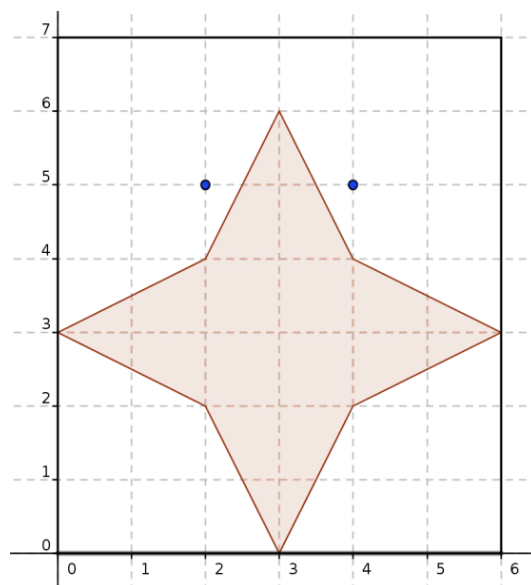
Выведите количество способов ударить мухобойкой так, чтобы ни одна муха не пострадала.

Примеры

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

Замечание

Пояснение к третьему примеру:



Задача G. Пришелец

Имя входного файла: `convexoid.in`
Имя выходного файла: `convexoid.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

В Спрингфилде появления НЛО — частое явление. Большинство летающих тарелок видели около атомной электростанции. Один из инопланетян даже живет в офисе Гомера. Или это мутировавшая жвачка Барта? Неважно...

Гомер еще не осознал, что пришелец разумный, и поэтому он устраивает эксперименты с ним. Гомер уже знает, что пришелец — это выпуклый двухсторонний многоклеточный многоугольник. Также он знает, что на i -й день пришелец вырастает на вектор V_i , то есть, если S — множество точек тела инопланетянина до i -го дня, то его тело после i -го дня будет следующим множеством точек:

$$S' = \{P + t \cdot V_i \mid P \in S, t \in [0, 1]\}.$$

Теперь Гомер хочет узнать его площадь после каждого из M дней.

Формат входных данных

Первая строка содержит два целых числа N и M . ($3 \leq N \leq 10^5$, $1 \leq M \leq 10^5$). Пришелец состоит из N вершин. Следующие N строк описывают его вершины в порядке обхода по или против часовой стрелки ($|X_i|, |Y_i| \leq 2^{27}$). Следующие M строк описывают вектора V_j . Координаты векторов не превосходят 2^{14} по абсолютной величине.

Формат выходных данных

Для каждого из следующих M дней выведите одно число — площадь пришельца после каждого дня. Помните, что пришелец двухсторонний!

Примеры

<code>convexoid.in</code>	<code>convexoid.out</code>
3 2	5
0 0	11
1 0	
0 1	
1 1	
1 -1	

Задача Н. Циклический сдвиг

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

Дана строка s , состоящая из маленьких латинских букв.

Назовем строку $t = t_1t_2 \dots t_m$ ($m > 0$) хорошей относительно строки s , если строка t и ее левый циклический сдвиг $t' = t_2 \dots t_mt_1$ являются подстроками строки s .

Вам необходимо найти количество различных хороших строк t относительно заданной строки s .

Формат входных данных

В единственной строке входных данных записана строка s , состоящая из n ($1 \leq n \leq 300\,000$) маленьких латинских букв.

Формат выходных данных

Выведите единственное число — количество хороших строк t относительно заданной строки s .

Примеры

стандартный ввод	стандартный вывод
abaac	7
aaa	3

Замечание

В первом примере хорошими строками являются следующие: a, b, c, aa, ab, ba, aba.

Во втором примере хорошими являются только три строки: a, aa, aaa.