

Задача А. Полёт хомяка 3

Имя входного файла: a.in
 Имя выходного файла: a.out
 Ограничение по времени: 1 с
 Ограничение по памяти: 64 Мб

В Хомякбурге проводится ежегодное соревнование летающих хомяков. В этом году правила следующие. Хомяка запускают из рогатки из некоторой точки на земле с начальной скоростью V . В воздухе находятся несколько контрольных точек. Если траектория полета хомяка пересекается с некоторой контрольной точкой, он останавливается в этой точке и снова может быть запущен из этой точки под любым углом с начальной скоростью V . Все контрольные точки находятся в одной плоскости, перпендикулярной земле. В той же плоскости находится стартовая точка. Также в воздухе расположены препятствия в виде кругов в той же плоскости. Траектория полета хомяка не может проходить через внутреннюю часть какого-либо круга (хотя может касаться круга). Сами круги не касаются и не покрывают контрольных точек и стартовой точки. Но могут пересекаться между собой и частично уходить под землю. Хомяк должен за определенное время T достичь заданной контрольной точки, прыгая по другим контрольным точкам. Если хомяк может это сделать, он получит Q очков. Чем меньше очков, тем лучше полет. Пусть хомяк прыгал по точкам p_0, p_1, \dots, p_k , где p_0 — стартовая точка, а p_k — целевая. Пусть он прилетел в точку p_i под углом a_i градусов, а вылетел под углом b_i , и пусть минимальный поворот от a_i до b_i равен по модулю c_i градусов. Тогда, для такого полета $Q = \max\{c_i\}$. Какое минимальное количество очков может получить хомяк? Размерами хомяка и рогаток, а также сопротивлением воздуха можно пренебречь. Ускорение свободного падения можно принять за 10 м/с^2 . По правилам хомяк не может начинать и заканчивать прыжок в одной и той же контрольной точке.

Формат входного файла

Первая строка содержит числа n — количество контрольных точек, m — количество кругов, V — скорость (м/с), T — время (с). Далее в n строках идут координаты каждой контрольной точки x, y (м). Хомяк должен достичь контрольной точки указанной последней. В следующих m строках идут координаты и радиусы каждого круга x, y, r (м). Стартовая точка имеет координаты $(0, 0)$.

Ограничения

$1 \leq n \leq 100$
 $0 \leq m \leq 100$
 $1 \leq V \leq 100$
 $1 \leq T \leq 100$
 $-100 \leq x \leq 100$
 $0 \leq y \leq 100$
 $1 \leq r \leq 100$

Формат выходного файла

Выведите минимальное количество очков, которое может получить хомяк с тремя знаками после запятой или -1 , если хомяк не сможет достичь контрольной точки за время T .

Пример

a.in	a.out
3 1 10 5 1 3 -1 2 4 0 1 1 1	44.310

Задача В. Интервалы

Имя входного файла: `b.in`
Имя выходного файла: `b.out`
Ограничение по времени: 2 с
Ограничение по памяти: 64 Мб

На вещественной прямой отмечено n интервалов. Концы каждого интервала попадают в целые точки. При этом концы интервалов не принадлежат им. Известно, какие отрезки пересекаются, а какие нет. По этой информации необходимо определить, какая может быть минимальная суммарная длина всех отрезков. Также интервалы не могут иметь нулевую длину (т.е. длина каждого из них должна быть хотя бы 1).

Формат входного файла

В первой строке входа даны числа n количество интервалов, и m количество пересечений между интервалами. Далее в m строках указаны пары пересекающихся интервалов. Интервалы нумеруются числами от 0 до $n - 1$. Если какая-то пара интервалов не указана, значит они не пересекаются. Гарантируется, что по данной информации можно построить набор интервалов соответствующий ей.

Ограничения

$$1 \leq n \leq 1234$$

$$0 \leq m \leq n(n - 1)/2$$

Формат выходного файла

Выведите минимально возможную суммарную длину всех интервалов. Интервалы не могут иметь нулевую длину.

Пример

<code>b.in</code>	<code>b.out</code>
3 2 0 1 1 2	4

Задача С. Иерархия

Имя входного файла: `c.in`
 Имя выходного файла: `c.out`
 Ограничение по времени: 3 с
 Ограничение по памяти: 64 Мб

В некоторой организации работает n сотрудников. Будем считать, что сотрудник A является подчиненным сотрудника B если:

1. A — непосредственный подчиненный B .
или
2. Существует сотрудник C , такой что A непосредственный подчиненный C , и C подчиненный B .

Ни один сотрудник не является собственным подчиненным. Различные сообщения в организации передаются от A к B только в том случае, если A подчиненный B , или наоборот. При этом отношения в организации устроены так, что сообщения могут быть переданы от любого сотрудника к любому другому. Необходимо для данных сотрудников A и B определять является ли A подчиненным B , или B подчиненным A , или отношение подчиненности между этими сотрудниками не установлено.

Формат входного файла

В первой строке входа дано количество сотрудников n и количество запросов q . Во второй и третьей строках даны параметры запросов X, a, b и Y, c, d . Далее в n строках идет описание структуры организации. Для i -того сотрудника в $(i+3)$ -й строке дана информация о его непосредственных подчиненных в отдельной строке: k_i — количество непосредственных подчиненных, а затем перечислены номера этих k непосредственных подчиненных. Сотрудники пронумерованы числами от 0 до $n-1$. Запросы имеют вид (x_j, y_j) , ответ на каждый запрос должен быть 1, если y_j является подчиненным x_j ; -1, если x_j является подчиненным y_j ; 0 — иначе. Здесь $x_0 = X, y_0 = Y, x_j = (ax_{j-1} + b + \text{sum}_{j-1}) \bmod n, y_j = (cy_{j-1} + d + \text{sum}_{j-1}) \bmod n, j = 1 \dots q-1$, а sum_{j-1} равно сумме модулей ответов на первые $j-1$ запросов.

Ограничения

$$1 \leq n \leq 10^5$$

$$1 \leq q \leq 5 \cdot 10^5$$

$$0 \leq X, a, b, Y, c, d < n$$

$$n-1 \leq \sum_{i=0}^{n-1} k_i < n+100$$

Формат выходного файла

Для каждого запроса выведите ответ в отдельной строке как описано выше.

c.in	c.out
6 3	1
1 3 5	0
4 1 4	-1
1 5	
3 0 2 3	
2 0 4	
1 4	
0	
2 3 4	

Задача D. Двоичные произведения

Имя входного файла: `d.in`
Имя выходного файла: `d.out`
Ограничение по времени: 1 с
Ограничение по памяти: 64 Мб

Есть последовательность из n единиц. Рассмотрим все различные способы вставить между ними одну и более операций умножения так, чтобы получившееся выражение имело смысл (т.е. операции умножения не должны стоять подряд, в начале и в конце последовательности). Например, для $n = 4$ есть 7 способов: 1×111 , 11×11 , 111×1 , $1 \times 1 \times 11$, $1 \times 11 \times 1$, $11 \times 1 \times 1$ и $1 \times 1 \times 1 \times 1$. Посчитаем сумму получившихся произведений, считая, что числа записаны в двоичной системе счисления: $1_2 \times 111_2 + 11_2 \times 11_2 + 111_2 \times 1_2 + 1_2 \times 1_2 \times 11_2 + 1_2 \times 11_2 \times 1_2 + 11_2 \times 1_2 \times 1_2 + 1_2 \times 1_2 \times 1_2 \times 1_2 = 33$.

Вы должны посчитать эту сумму для произвольного n .

Ограничения

$$2 \leq n \leq 10^9$$

Формат входного файла

Первая строка содержит T ($1 \leq T \leq 1000$) — количество тестов. Следующие T строк содержат n .

Формат выходного файла

Для каждого n из исходных данных программа должна выдать искомую сумму по модулю 1000000007.

Пример

<code>d.in</code>	<code>d.out</code>
2	1
2	33
4	

Задача Е. Упаковочная машина

Имя входного файла: `e.in`

Имя выходного файла: `e.out`

Ограничение по времени: 1 с

Ограничение по памяти: 64 Мб

В упаковочную машину загрузили n игрушечных Дедов Морозов и n Снегурочек. Машина может упаковывать:

1. одного Деда Мороза в упаковку одного из двух видов;
2. одну Снегурочку;
3. Деда Мороза и Снегурочку в одну упаковку.

Таким образом, после того, как все игрушки будут упакованы, на выходе машины будет последовательность из упаковок 4-х видов. Сколько таких различных последовательностей может сделать машина?

Ограничения

$$1 \leq n \leq 10^6$$

Формат входного файла

Первая строка входного файла содержит целое число n .

Формат выходного файла

Выведите единственное целое число — количество различных последовательностей упаковок по модулю 1000003.

Пример

<code>e.in</code>	<code>e.out</code>
1	5
2	37