

Задача А. Количество линий

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

Известный программист Петя планирует создать совершенно новую увлекательную компьютерную игру, которая не снилась никому из ведущих фирм, занимающихся разработкой игр. Петя уверен, что благодаря тем революционным особенностям, которые он намерен реализовать, его игра быстро завоюет рынок и принесет ему всемирную славу и баснословные прибыли.

Одной из таких особенностей будет система отображения результатов в таблице рекордов. Каждый результат – это число, записываемое в десятичной системе. Но цифры и знак минус будут отображаться с помощью нескольких линий:



Помогите Пете узнать сколько линий его программа нарисует, когда игрок наберет определенное количество очков.

Формат входных данных

В единственной строке задается одно целое число X ($-10^9 \leq X \leq 10^9$), определяющее количество очков, набранных игроком.

Формат выходных данных

В единственную строку выведите одно число – количество линий, необходимых для изображения результата.

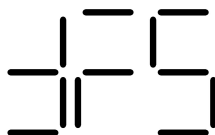
Примеры

<code>count.in</code>	<code>count.out</code>
85	12
-3	6

Задача В. Восстановление количества очков

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

Не менее известный хакер Вася, увидев замечательную игру Пети, решил немного испортить ее. В подпрограмме вывода количества очков участника он подправил код таким образом, что каждая из линий, которая используется в изображении, может быть выведена на экран, а может быть и пропущена. Таким образом, игрок не всегда мог ясно понять сколько у него очков. Например, число 325 могло бы быть изображено как



Но ведь таким же образом могло быть записано и число 986, и еще несколько чисел. Ваша задача – по заданному изображению определить сколько целых чисел, определяющих количество очков, могло быть записано таким образом.

Формат входных данных

В первой строке задается число K ($1 \leq K \leq 9$) – количество символов в числе, которое нужно было отобразить. В каждой из следующих K строк записаны по 7 цифр, каждая из которых либо 0, либо 1. Эти цифры соответствуют линиям в изображении соответствующего символа числа, перечисленным сверху вниз, а на одном уровне слева направо. То есть первая цифра соответствует самой верхней линии, вторая – верхней левой, третьей – верхней правой, четвертая – средней, пятая – нижней левой, шестая – нижней правой, седьмая – самой нижней. Значение 1 обозначает наличие линии, а 0 – отсутствие. Изображение задается слева направо, то есть первым дается описание изображение самого левого символа, использованного в записи числа, последним – самого правого.

Формат выходных данных

В единственную строку выведите одно число – количество чисел, которые могли быть изображены заданным образом. Учтите, что при выводе какого-либо символа могло быть не отображено ни одной линии, однако при правильном выводе ни одно знакоместо не должно было оказаться пустым.

Примеры

<code>guess.in</code>	<code>guess.out</code>
3 0011011 1001100 1101011	36
1 1110111	2
2 0000000 0000000	99

Задача С. Секретный уровень

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

Поиграв достаточное время в игру Пети, Коля обнаружил, что в ней есть секретный уровень, для получения доступа к которому нужно ввести очень секретный код и очень долго ждать проверки его правильности. Коля даже однажды видел и запомнил код, который вводил Петя, играя в свою игру, но, попытавшись ввести самостоятельно этот код, получил сообщение о том, что он неверный. Поскольку уровень обещал быть невероятно богат разными плюшками, Коля решил, что должен непременно на него попасть и обратился за помощью к хакеру Васе. После долгого изучения кода игры Пети Васе так и не удалось снять защиту с этого уровня, но удалось выяснить как можно получить этот код.

Способ получения кода заключается в следующем. Берется число, равное серийному номеру компьютера, на котором запущена игра. Далее каждая цифра этого числа умножается на саму себя, полученные при этом числа записываются рядом друг с другом, образуя новое число. Эта операция выполняется столько раз, сколько было перезагрузок компьютера в течение дня. Кодом будет количество цифр в числе, которое получится в итоге.

Формат входных данных

В единственной строке задаются два целых числа: количество перезагрузок k и серийный номер компьютера Коли N ($0 \leq k \leq 100$, $0 \leq N \leq 10^{13}$).

Формат выходных данных

В единственную строку выведите одно число – код для секретного уровня в игре Пети.

Примеры

	<code>level.in</code>	<code>level.out</code>
1	85	4
2	4	3

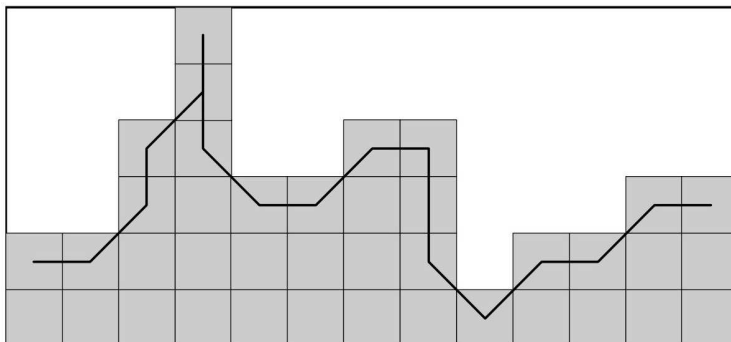
Замечание

В первом примере, после возведения в квадрат каждой цифры, получается число 6425, состоящее из четырех цифр. Во втором примере, сначала 4 возводится в квадрат – получается 16, а после возведения в квадрат цифр числа 16 получится трехзначное число 136.

Задача D. Горная гряда

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

Во время сражений в игре Пети на дальнем плане изображается горная гряда, которая представляет собой связанное и выпуклое в вертикальном направлении множество пикселей, нижняя сторона которого параллельна горизонтали.



Поскольку, как сказал Вася, эта гряда генерируется с помощью датчика случайных чисел, который потом используется и в процессе боя, Коля уверен, что изучив внимательно горы, он сможет предугадывать ход боя. В частности, его интересуют протяженность горной гряды слева направо (количество пикселей на нижней стороне) и максимальный перепад высот (расстояние по вертикали между самым верхним и самым нижним пикселем).

Формат входных данных

Гряда задается своей верхней огибающей ломаной, каждое звено которой соединяет соседние пиксели по горизонтали, вертикали или диагонали. В первой строке задается целое число N ($1 \leq N \leq 100000$), количество звеньев ломаной. В каждой из последующих N строк задается пара чисел x и y – координаты вектора, определяющего соответствующее звено ломаной ($0 \leq x \leq 1$, $-1 \leq y \leq 1$, $x^2 + y^2 \neq 0$).

Формат выходных данных

В единственную строку выведите два целых числа – протяженность горной гряды и максимальный перепад высот.

Пример

<code>mountain.in</code>	<code>mountain.out</code>
18	13 5
1 0	
1 1	
0 1	
1 1	
0 1	
0 -1	
0 -1	
1 -1	
1 0	
1 1	
1 0	
0 -1	
0 -1	
1 -1	
1 1	
1 0	
1 1	
1 0	

Задача Е. Караваны

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

Ну и конечно же Петя не мог при разработке не учесть пожеланий мирового геймерского сообщества, поэтому в его игре можно “набегать” и “грабить караваны”. Караван состоит из повозок, в каждой из которых находится определенное количество золота. Ограбление можно начать с любой повозки, продвигаясь далее к следующей или предыдущей и забирая золото, которое находится в них. Всю операцию нужно провести достаточно быстро, иначе подоспеет охрана и схватит грабителей. Поэтому забрать можно будет не из всех повозок.

Коля, играющий, как и следовало ожидать, за эльфов, знает и количество повозок в караване N , и сколько золота в каждой повозке (в i -ой повозке находится a_i золота), и максимальное количество повозок k , которое его отряд успеет обойти до прибытия охраны. Ему необходимо узнать максимальное количество золота, которое он может получить в результате операции.

Формат входных данных

В первой строке заданы два целых числа N и k ($1 \leq k \leq N \leq 100000$), определяющие размер каравана и максимальное число повозок, которые можно ограбить. Во второй строке задаются N целых чисел a_i , определяющие количество золота в повозках ($0 \leq a_i \leq 10000$).

Формат выходных данных

В единственной строке выведите наибольшую сумму, которую может принести ограбление.

Пример

<code>caravan.in</code>	<code>caravan.out</code>
7 3 3 2 1 2 3 2 1	7