

## Задача А. Сигнальные башни

Имя входного файла: `beacon.in`  
Имя выходного файла: `beacon.out`  
Ограничение по времени: 0,5 секунды

На северной границе королевства построено в одну линию  $N$  сигнальных башен, на которых при приближении врага зажигают костры. Костёр, зажжённый на башне  $A$ , увидят на башне  $B$  ( $1 \leq A, B \leq N$ ), если эти башни стоят рядом или между ними нет более высоких башен, чем самая высокая из башен  $A$  или  $B$ . Так, если имеется семь башен и их высоты равны соответственно 2, 4, 1, 2, 2, 5 и 1, то костёр, зажжённый на второй башне, будет замечен на всех других башнях, кроме последней. Огонь, горящий на первой или последней башне, будет виден только с соседней башни.

Найдите количество пар  $(A, B)$ , где  $A < B$ , для которых огонь, зажжённый на башне  $A$ , будет виден с башни  $B$ , и наоборот.

### Формат входного файла

Первая строка содержит величину  $N$  ( $2 \leq N \leq 500\,000$ ). Далее следуют  $N$  строк, каждая из которых содержит одно целое положительное число, не превосходящее  $2^{31} - 1$ , — высоту очередной башни.

### Формат выходного файла

Выведите единственное число — искомое количество пар башен.

### Примеры

<code>beacon.in</code>	<code>beacon.out</code>
7 2 4 1 2 2 5 1	10

## Задача В. Книжная полка

Имя входного файла: `bookshelf.in`  
Имя выходного файла: `bookshelf.out`  
Ограничение по времени: 2 секунды

Саша недавно переехал в новую комнату общежития и сейчас занимается распаковкой вещей из огромного числа коробок. Как и многие другие студенты ФПМИ БГУ, он считает, что даже на книжной полке необходим порядок и книги следует размещать в порядке возрастания высоты (т. е. на левой стороне полки должна стоять самая низкая книга, а при просмотре слева направо высота книг должна увеличиваться или не изменяться).

Но Саша ещё и очень суеверный студент, поэтому он берёт книги из коробки по одной и сразу ставит их на полку, не двигая другие книги. Он может поставить книгу около левой стенки полки, если там пока нет книги, либо справа вплотную к книгам, касающимся левой стенки. Аналогично, Саша может разместить книгу прямо у правой стенки полки или же приставить книгу слева к стоящим в правой стороне книгам, если место непосредственно у правой стенки занято. Саша решил, что некоторые книги можно не ставить на полку, а сразу убрать в шкаф. Пусть пылятся!

Саша знает, в каком порядке он будет доставать книги из коробки. Помогите ему поставить на полку как можно больше книг. Полка достаточно широкая, чтобы все книги могли на ней поместиться по ширине. Но помните про порядок на полке и суеверия Саши!

### Формат входного файла

В первой строке входного файла записано единственное целое число  $N$  ( $1 \leq N \leq 100$ ) — количество книг в коробке.

Во второй строке записаны  $N$  натуральных чисел.  $i$ -е число задает высоту книги, которую достанет  $i$ -й по счету из коробки Саша. Все числа во входном файле не превосходят 100.

### Формат выходного файла

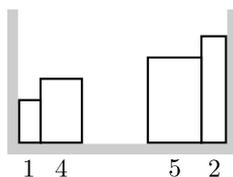
В единственной строке выходного файла выведите наибольшее количество книг, которое можно поставить на полку.

### Примеры

bookshelf.in	bookshelf.out
5 2 5 1 3 4	4
8 1 8 3 6 5 4 7 2	6

### Примечание

В первом примере Саша может выставить на полку книги так, чтобы получить набор высот (2345) или (1345).



Чтобы получить первый набор, Саша, извлекая из коробки первую книгу высоты 2, ставит её у левой стенки полки. Затем вторую книгу (высоты 5) ставит у правой стенки. Следующая книга отправляется в шкаф, потому что на полке её следовало бы поставить левее первой книги, которая уже вплотную прилегает к левой стенке и не может быть подвинута. Четвёртая книга (высоты 3) ставится на левую сторону полки справа от первой книги. Последнюю, пятую, книгу можно либо разместить на левой стороне (правее четвёртой книги), либо на правой стороне (левее второй книги).

## Задача С. Конфеты

Имя входного файла: `candy.in`  
Имя выходного файла: `candy.out`  
Ограничение по времени: 2 секунды

У Кэнди есть запас конфет  $F$  разных сортов. Она собирается упаковать эти конфеты в несколько коробок на продажу. Каждая получившаяся коробка должна быть одного из двух типов:

- либо это коробка, которая содержит конфеты только одного сорта;
- либо коробка, которая содержит конфеты каждого из сортов.

Кэнди хочет, чтобы распределение конфет по коробкам было красивым, и её чувство прекрасного требует соблюдать следующие условия.

- Каждая конфета должна быть помещена ровно в одну коробку.
- Каждая коробка, независимо от её типа, должна содержать по крайней мере две конфеты.
- Каждая коробка, независимо от её типа, должна содержать одинаковое число конфет.
- Если коробка содержит несколько сортов, то количество конфет каждого сорта должно быть одинаковым.
- Должна быть, как минимум, одна коробка с конфетами каждого из сортов.
- Для каждого из сортов должна быть, как минимум, одна коробка с конфетами только этого сорта.

Кэнди очень любопытно, сколько различных вариантов красивой упаковки конфет она может получить. Два красивых варианта считаются различными тогда и только тогда, когда они различаются либо количеством коробок с конфетами одного сорта, либо количеством коробок с конфетами разных сортов, либо количеством конфет в коробках. Помогите ей найти количество различных вариантов красивой упаковки.

### Формат входного файла

Первая строка каждого теста содержит целое число  $F$  — количество имеющихся сортов конфет ( $2 \leq F \leq 10^5$ ). Вторая строка содержит  $F$  целых чисел  $C_i$ , разделенных пробелами, — количество имеющихся конфет каждого из сортов ( $1 \leq C_i \leq 10^9$ ,  $1 \leq i \leq F$ ).

### Формат выходного файла

Необходимо вывести единственное целое число — количество разных красивых вариантов упаковки конфет согласно правилам, описанным выше.

### Примеры

<code>candy.in</code>	<code>candy.out</code>
3 15 33 21	4
2 1 1	0
2 2 2	0
2 3 3	1
3 1000000000 1000000000 1000000000	832519396

## Задача D. Лягушка-путешественница

Имя входного файла: frog.in  
Имя выходного файла: frog.out  
Ограничение по времени: 1 секунда

Лягушка-путешественница очень любит знакомиться с миром, передвигаясь на гусях-лебедях. Она уже не раз облетала окрестности и вполне уверена, что не найдёт больше ничего интересного, кроме  $n$  болот, расположенных в линию на расстоянии  $\ell$  вёрст между соседними, причём живёт она на краю этого скучного мира, то есть в одном из двух самых удалённых друг от друга болот. Поэтому очередной отпуск Лягушка решила провести, наслаждаясь любимыми местами и дружным кваканьем со старыми знакомыми. Она собирается пролететь  $m$  раз от своего болота к любимому, самому далёкому от своего, болоту и обратно и хочет посетить каждое из  $n$  болот, кроме разве что одного, хотя бы по одному разу. При этом гуси-лебеди не могут преодолевать расстояние более  $3\ell$  вёрст за один перелёт, поэтому Лягушка будет вынуждена делать остановки достаточно часто, но всё же не собирается менять направление движения в любом болоте, кроме своего и любимого.

И вдруг Лягушке стало интересно, как долго она сможет проводить отпуск подобным образом, если она не хочет, чтобы один был похожим на другой. Похожими она будет считать два отпуска, если порядок остановок между перелётами полностью совпадает.

### Формат входного файла

Единственная строка содержит два целых числа  $m$  ( $1 < m \leq 10^{12}$ ) и  $n$  ( $1 < n \leq 10$ ) — количество двухсторонних перелётов и количество болот.

### Формат выходного файла

Единственная строка должна содержать остаток от деления числа способов провести отпуск на 1 000 000 000.

### Примеры

frog.in	frog.out
4 4	65535

## Задача Е. Драгоценные камни

Имя входного файла: `jewelry.in`  
Имя выходного файла: `jewelry.out`  
Ограничение по времени: 0,5 секунды

Вы являетесь счастливым обладателем семи драгоценных камней из древней коллекции. Коллекция интересна тем, что визуально все камни в ней одинаковы, но могут иметь различный вес.

Вы часто передаёте свои сокровища для демонстрации на всевозможных выставках. И вот однажды случилась беда — организаторы выставки по небрежности сначала не записали вес каждого камня, а затем положили Ваши камни рядом с двумя другими, такими же по виду, но принадлежащими другому человеку.

Сможете ли Вы, взвесив каждый из девяти камней, отделить Ваши камни от чужих? Вам известно лишь, что суммарный вес Ваших камней равен 100.

### Формат входного файла

Входной файл содержит девять строк, в каждой из которых записано целое положительное число, не превосходящее 100, — вес очередного камня. Тесты подобраны таким образом, что существует ровно одно решение задачи.

### Формат выходного файла

Выведите в выходной файл семь строк — веса Ваших камней в том порядке, в котором они записаны во входном файле.

### Примеры

<code>jewelry.in</code>	<code>jewelry.out</code>
7	7
8	8
10	10
13	13
15	19
19	20
20	23
23	
25	
8	8
6	6
5	5
1	1
37	30
30	28
28	22
22	
36	

## Задача F. Ровно K

Имя входного файла: `paths.in`  
Имя выходного файла: `paths.out`  
Ограничение по времени: 2 секунды

Вася — большой любитель настольных игр. Недавно он приобрёл новую игру для одного игрока с интригующим названием «Ровно K» — это продолжение популярной серии игр «Ровно 1», «Ровно 2», «Ровно 3»...

На игровом поле отмечено  $N$  позиций, которые пронумерованы натуральными числами от 1 до  $N$ . Фишка игрока первоначально располагается на первой позиции. Некоторые пары позиций соединены линиями. По правилам игры за один ход фишку нужно обязательно переместить из текущей позиции в другую, соединённую с текущей. Если это сделать невозможно, то игра заканчивается и игроку засчитывается поражение. Между двумя позициями может быть проведено не более одной линии. Бывают, однако, линии, которые ведут из позиции в неё же, тогда фишка в течение одного хода может оставаться неподвижной. Все соединения действуют в обе стороны: если позиция  $i$  соединена с позицией  $j$ , то фишку можно передвигать как из позиции  $i$  в позицию  $j$ , так и из позиции  $j$  в  $i$ .

Игра состоит из нескольких независимых партий. Каждая партия описывается двумя числами:  $V$  и  $K$ . Чтобы выиграть партию, игроку нужно переместить фишку из позиции 1 в позицию  $V$ , сделав ровно  $K$  ходов.

Вася распаковал коробку, разложил на столе игровое поле, извлёк брошюру с правилами игры и списком из  $Q$  пар чисел — это числа  $V$  и  $K$  для каждой партии. Какие партии Вася сможет выиграть при оптимальной игре, а какие выиграть принципиально невозможно?

### Формат входного файла

В первой строке входного файла записаны два целых числа  $N$  ( $1 < N \leq 5000$ ) и  $M$  ( $0 \leq M \leq 500\,000$ ) — количество позиций на игровом поле и количество соединительных линий. Последующие  $M$  строк описывают соединения: линия задаётся парой номеров позиций, которые она соединяет. Затем в отдельной строке записано число партий  $Q$  ( $1 \leq Q \leq 5000$ ), которые предстоит сыграть Васе. Затем в  $Q$  строках описаны партии: каждая пара чисел  $V$  и  $K$  ( $1 \leq V \leq N$ ,  $0 \leq K \leq 1\,000\,000\,000$ ) задана в отдельной строке.

### Формат выходного файла

Выходной файл должен содержать ровно  $Q$  строк, по одной для каждой партии. В случае, если существует путь фишки из позиции 1 в позицию  $V$  из ровно  $K$  ходов, выведите в соответствующей строке Yes, иначе выведите No.

### Примеры

<code>paths.in</code>	<code>paths.out</code>
6 8	Yes
1 2	Yes
1 4	No
3 4	
2 5	
4 6	
5 6	
4 5	
3 6	
3	
3 2	
6 5	
6 1	

## Задача G. Головоломка

Имя входного файла: puzzle.in  
Имя выходного файла: puzzle.out  
Ограничение по времени: 1 секунда

Конечно же, Вы знаете головоломку «15»! В коробочку размера  $4 \times 4$  помещаются 15 квадратных плиток размера  $1 \times 1$ , и на каждой плитке изображено число от 1 до 15. Одно место в коробке остаётся свободным (будем рисовать на этом месте 0). Передвигая на свободное место одну из соседних плиток, требуется собрать их в порядке возрастания написанных чисел, а свободное место оставить в правом нижнем углу. Иными словами, требуется получить следующую конфигурацию плиток:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	0

Рассмотрим усложнённый вариант этой головоломки, когда коробочка имеет размер  $N \times N$  и в неё помещено  $N^2 - 1$  плиток с соответствующими числами.

Вы заняты написанием компьютерного варианта этой головоломки. В частности, Вам поручено сгенерировать начальное расположение плиток. При этом Вы желаете оценить *сложность* этого расположения. Под сложностью Вы понимаете сумму расстояний между начальной и конечной позицией каждой плитки, вычисленных по формуле Минковского. Так, если изначально плитка стоит в позиции  $(x_1, y_1)$ , а в решённой головоломке она должна стоять в позиции  $(x_2, y_2)$ , то расстояние между начальной и конечной позицией для этой плитки равно  $|x_1 - x_2| + |y_1 - y_2|$ . Расстояние для пустой позиции считается по этой же формуле.

Сможете ли Вы рассчитать сложность сгенерированного начального расположения плиток?

### Формат входного файла

Первая строка входного файла содержит значение  $N$  ( $4 \leq N \leq 250$ ). Каждая из последующих  $N$  строк содержит  $N$  целых неотрицательных чисел — изображение, помещённое на соответствующей плитке. Все числа различны и не превосходят  $N^2 - 1$ .

### Формат выходного файла

Выведите в выходной файл строку с единственным числом — рассчитанной сложностью начального расположения плиток.

### Примеры

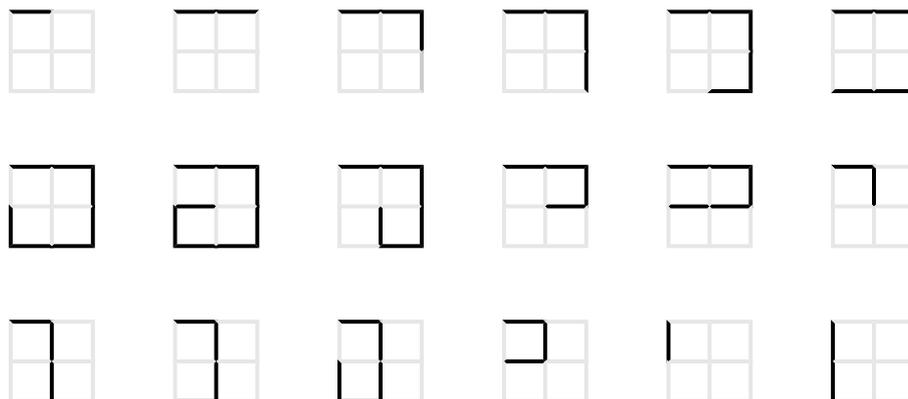
puzzle.in	puzzle.out
4 1 2 3 4 5 6 7 8 9 10 11 12 13 0 14 15	4
4 8 11 4 10 9 1 12 7 2 13 15 3 0 5 6 14	38

## Задача Н. Спирали

Имя входного файла: `spiral.in`  
Имя выходного файла: `spiral.out`  
Ограничение по времени: 2 секунды

Имеется клетчатое поле размера  $N \times M$ . Начиная с верхнего левого угла поля, нарисуем по клеткам спираль — ломаную без самопересечений и самокасаний, которая закручена по часовой стрелке (при движении вдоль ломаной все повороты, если они есть, осуществляются на угол  $90^\circ$  и только вправо). Будем полагать, что отрезок также является спиралью, однако одна точка (левый верхний угол) сама по себе спиралью не является.

Сколько всего можно нарисовать различных спиралей на данном поле?



### Формат входного файла

В первой и единственной строке входного файла содержится два целых числа  $N$  и  $M$  ( $1 \leq N, M \leq 5000$ ) — размер поля. Числа разделены одиночным пробелом.

### Формат выходного файла

Выведите ровно одно число — количество различных спиралей, которые можно изобразить на поле заданного размера, по модулю 1 000 000 007.

### Примеры

<code>spiral.in</code>	<code>spiral.out</code>
1 1	4
2 2	18

### Примечание

На рисунке изображены всевозможные спирали для поля  $2 \times 2$ .