

Задача А. Путешествие чатланина

Имя входного файла: `travel.in`
Имя выходного файла: `travel.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Mebibytes

В галактике Кин-Дза-Дза существовал, в частности, такой обычай. Когда на планету, населённую пацаками, прибывал пепелац с экипажем, состоящим из чатлан, специально выбранные для этой миссии N пацаков выстраивались в шеренгу по росту и приветствовали чатлан.

Так как процесс происходил спонтанно, то ошибки в построении по росту были довольно частыми. Если чатланин видел, что пацак, у которого рост выше, стоял левее, чем пацак, у которого рост ниже, то все пацаки должны были присесть и хором сказать «ку!».

Один чатланин облетел так несколько планет, и заинтересовался, сколько раз за полёт он слышал хоровое «ку!» от пацаков при построениях.

Формат входного файла

Первая строка входных данных содержит количество тестов. Каждый тест состоит из двух натуральных чисел $N \leq 10000$ — количество встречающихся пацаков, $M \leq 100$ — количество посещённых чатланином планет. Далее в следующих M строках — описание построений. Построение — это последовательность из N натуральных чисел разделённых пробелом, i -ое число — это рост пацака, стоящего на i -ом месте. Гарантируется, что все задействованные пацаки разного роста и рост самого высокого пацака не превосходит N .

Формат выходного файла

Для каждого теста программа должна вывести в выходной файл в отдельной строке количество хороших «ку!», услышанных чатланином за путешествие.

Пример

<code>travel.in</code>	<code>travel.out</code>
2	4
3 2	2
1 3 2	
3 2 1	
4 1	
1 4 2 3	

Задача В. Раздел пустыни

Имя входного файла: `desert.in`
Имя выходного файла: `desert.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Mebibytes

Господин ПэЖэ принял решение упорядочить гастрольную деятельность землян в пустыне планеты Плюк. Для этого он приказал разбить пустыню (представляющую собой прямоугольник размером $3 \times N$) на стандартные участки 1×2 , и каждый такой участок выделить культбригаде из одного чатланина, одного пацака и двух землян. При этом каждая точка пустыни должна или принадлежать ровно одному участку, или лежать на границе участков.

Сколькими различными способами можно осуществить приказ господина ПэЖэ?

Формат входного файла

Первая строка входных данных содержит количество тестов. Каждый тест записывается в отдельной строке и содержит единственное натуральное число $N \leq 32$.

Формат выходного файла

Для каждого теста программа должна вывести в выходной файл в отдельной строке число различных способов разбиения пустыни на участки.

Пример

<code>desert.in</code>	<code>desert.out</code>
2	11
4	3
2	

Задача С. Перепись пацаков

Имя входного файла: `line.in`
Имя выходного файла: `line.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Mebibytes

На планете Плюк объявлена перепись пацаков. Перепись происходит следующим образом. Первый переписчик начинает работу в 8.00, второй — в 8.05. На одного пацака переписчик тратит 10 минут (в основном — на подбор устраивающего того дизайна цака). Последний пацак может быть принят в 20.00, после чего пункт переписи закрывается. Очередь единая, и очередной пацак обслуживается, как только освобождается один из переписчиков. Гедеван прибыл на пункт переписи в 8.00 и увидел, сколько пацаков стоит перед ним. Требуется определить, сколько времени ему придется простоять в очереди, и вообще, перепишут ли его сегодня.

Формат входного файла

Первая строка входа содержит количество тестов N . Далее идет N строк, в каждой из которых записано целое число K ($1 \leq K \leq 250$) — номер Гедевана в очереди.

Формат выходного файла

Для каждого теста в выходной файл выводится строка "NO если Гедеван сегодня уже не успевает к переписчику, и время его ожидания (в формате $X Y$, где X — количество целых часов, которые простоят в очереди Гедеван, и Y — количество минут), если все же успеет заплатить.

Пример

<code>line.in</code>	<code>line.out</code>
5	0 0
1	1 35
20	NO
235	5 45
70	NO
200	

Задача D. Гравицапа

Имя входного файла: gravitsapa.in
Имя выходного файла: gravitsapa.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Mebibytes

Гравицапа состоит из двух элементов. Верхний элемент может быть одного из N типов, нижний — одного из M типов. И хотя по выводам верхний и нижний элементы всегда совместимы между собой, некоторые типы верхних элементов не сочетаются с некоторыми типами нижних. Конструкция гравицапы полностью определяется типом верхнего и нижнего элементов.

На пункте технического обслуживания пепелацев имеется по одному верхнему элементу каждого типа и по одному нижнему элементу каждого типа. Сколько существует различных способов собрать из имеющихся элементов K гравицап?

Формат входного файла

Первая строка входных данных содержит количество тестов. Каждый тест содержит 3 целых числа: N — количество верхних элементов гравицапы, M — количество нижних элементов гравицапы и K — требуемое количество гравицап ($1 \leq N, M, K \leq 10$). Далее идёт таблица совместимости — N строк длины M . Если символ, стоящий в i -ой строке на j -ом месте равен Y , то i -й верхний элемент и j -й нижний элемент совместимы. Если этот символ равен N , то соответствующие элементы несовместимы.

Формат выходного файла

Для каждого теста программа должна вывести в выходной файл в отдельной строке число различных способов сборки ровно K гравицап.

Пример

gravitsapa.in	gravitsapa.out
2	24
3 4 3	3
YYYY	
YYYY	
YYYY	
1 5 1	
YYNNY	

Задача E. КЦ

Имя входного файла: `kc.in`
Имя выходного файла: `kc.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Mebibytes

В связи с экономическим кризисом банки планеты Плюк перешли на хранение депозитов в «мультивалютных коробках». Мультивалютная коробка — это коробка, в которой содержатся КЦ более, чем одного вида. При этом две такие коробки считаются одинаковыми, если содержат одинаковые виды КЦ.

Вычислите количество различных «мультивалютных коробок» для N типов КЦ.

Формат входного файла

Первая строка входных данных содержит количество тестов. Каждый тест содержит единственное натуральное число $N < 32$ — количество типов КЦ на планете.

Формат выходного файла

Для каждого теста программа должна вывести количество различных «мультивалютных коробок».

Пример

<code>kc.in</code>	<code>kc.out</code>
2	4
3	11
4	

Задача F. Дворец

Имя входного файла: palace.in
Имя выходного файла: palace.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Mebibytes

Дворец господина ПэЖэ несколько стоящих рядом башен из кубических блоков. Самая левая башня состоит из одного блока, а каждая следующая ровно на один блок выше предыдущей. Требуется по имеющемуся у строителей числу блоков определить, какова будет высота наиболее высокой башни в построенном из этих блоков дворце господина ПэЖэ.

Формат входного файла

Первая строка входа содержит количество тестов N . Каждый из тестов состоит из числа — количества имеющихся в наличии блоков ($1 \leq K \leq 100000$).

Формат выходного файла

Для каждого теста в выходной файл вывести строку, содержащую число — количество блоков в последней башне.

Пример

palace.in	palace.out
6	1
1	1
2	2
3	2
4	2
5	3
6	

Задача G. Ловушки

Имя входного файла: traps.in
Имя выходного файла: traps.out
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 Mebibytes

Далеко не все в Тентуре имеют право носить малиновые штаны, и конечно, не все владеют пепелацем с гравицапой, зато один предприимчивый чатланин поставил несколько ловушек, захватывающих отваливающиеся детали с пролетающих мимо инопланетных кораблей. Каждое утро он их обходит в надежде, что рано или поздно из этих запчастей ему удастся собрать какое-нибудь транспортное средство. Путь свой чатланин планирует по карте, на которой аккуратно отмечены ловушки, отсчитывая начало координат от собственного дома. Начинается обход от дома, в нем же и заканчивается. Требуется определить, какое расстояние проходит чатланин за день.

Формат входного файла

Первая строка входа содержит количество тестов N . Каждый из тестов состоит из $+1$ строк: в первой строке — количество ловушек, установленных чатланином ($0 < K \leq 0$), далее K строк формата $X Y$, описывающих координаты ловушек в порядке их обхода ($-10 \leq X, Y \leq 0$); X, Y — целые.

Формат выходного файла

Для каждого теста в выходной файл выводится общее расстояние, которое пройдет чатланин, обойдя все ловушки в приведенном порядке и вернувшись в начало координат от последней ловушки, с точностью до двух знаков после запятой.

Пример

traps.in	traps.out
3	2.00
1	3.41
1 0	4.00
2	
1 0	
1 1	
3	
1 0	
1 1	
0 1	