

Задача А. Федя и вопросы «Своей игры»

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Как известно, в ЛКШ.2013.Июль «Свою игру» проводит Федя. Только что обнаружилось, что для предстоящего турнира ему не хватает одной темы. Тема в «Своей игре» — это набор ровно из пяти вопросов, лежащих в одной области, причем каждый следующий вопрос строго сложнее предыдущего. К счастью, у Феде уже есть пакет из N готовых вопросов из области «Исторические личности Берляндии». Для каждого из вопросов известна его сложность (натуральное число). Посчитайте, сколько у Феде способов собрать из вопросов пакета одну тему. Два способа считаются различными, если хотя бы один вопрос, присутствующий в одном из них, не присутствует во втором.

Формат входных данных

В первой строке файла находится одно число N — количество вопросов в наборе Феде ($5 \leq N \leq 300$). В следующей строке N чисел, отсортированных по неубыванию — сложности вопросов из пакета. Каждая сложность — натуральное число от 1 до 100, сложности некоторых вопросов могут совпадать.

Формат выходных данных

Выведите одно число — количество способов сделать тему для «Своей игры».

Примеры

<code>stdin</code>	<code>stdout</code>
7 1 1 2 3 4 5 5	4
7 1 2 3 4 5 5 6	11
5 1 1 2 3 4	0

Задача В. Наибольшая общая подпоследовательность

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 3 секунды
Ограничение по памяти: 64 мегабайта

Даны две последовательности. Найдите длину их наибольшей общей подпоследовательности (подпоследовательность — это то, что можно получить из данной последовательности вычеркиванием некоторых элементов).

Формат входных данных

В первой строке входных данных записано целое число N — длина первой последовательности ($1 \leq N \leq 1000$). Во второй строке записаны N чисел — члены первой последовательности. В третьей строке записано целое число M — длина второй последовательности ($1 \leq M \leq 1000$). В четвертой строке записаны M чисел — члены второй последовательности. Члены последовательностей — целые числа, не превосходящие 10 000 по модулю.

Формат выходных данных

Программа должна вывести единственное целое число — длину наибольшей общей подпоследовательности, или число 0, если такой не существует.

Примеры

<code>stdin</code>	<code>stdout</code>
3 1 2 3 4 2 1 3 5	2

Задача С. Калькулятор

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 4 секунды
Ограничение по памяти: 64 мегабайта

Имеется калькулятор, который выполняет следующие операции:

- умножить число X на 2;
- умножить число X на 3;
- прибавить к числу X единицу.

Определите, какое наименьшее количество операций требуется, чтобы получить из числа 1 число N .

Формат входных данных

Во входном файле написано натуральное число N , не превосходящее 10^6 .

Формат выходных данных

В первой строке выходного файла выведите минимальное количество операций. Во второй строке выведите числа, последовательно получающиеся при выполнении операций. Первое из них должно быть равно 1, а последнее N . Если решений несколько, выведите любое.

Примеры

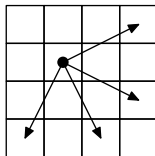
<code>stdin</code>	<code>stdout</code>
1	0 1
5	3 1 3 4 5
962340	17 1 3 9 27 54 55 165 495 1485 4455 8910 17820 17821 53463 160389 160390 481170 962340

Задача D. Ход конём - 2

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Дана прямоугольная доска $N \times M$ (N строк и M столбцов). В левом верхнем углу находится шахматный конь, которого необходимо переместить в правый нижний угол доски.

При этом конь может ходить следующим образом:



Необходимо определить, сколько существует различных маршрутов, ведущих из левого верхнего в правый нижний угол.

Формат входных данных

Входной файл содержит два натуральных числа N и M ($1 \leq N, M \leq 50$).

Формат выходных данных

В выходной файл выведите единственное число — количество способов добраться конём до правого нижнего угла доски.

Примеры

<code>stdin</code>	<code>stdout</code>
4 4	2
2 3	1

Задача Е. Покупка билетов

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

За билетами на премьеру нового мюзикла выстроилась очередь из N человек, каждый из которых хочет купить 1 билет. На всю очередь работала только одна касса, поэтому продажа билетов шла очень медленно, приводя «постояльцев» очереди в отчаяние. Самые сообразительные быстро заметили, что, как правило, несколько билетов в одни руки кассир продаёт быстрее, чем когда эти же билеты продаются по одному. Поэтому они предложили нескольким подряд стоящим людям отдавать деньги первому из них, чтобы он купил билеты на всех.

Однако для борьбы со спекулянтами кассир продавала не более 3-х билетов в одни руки, поэтому договориться таким образом между собой могли лишь 2 или 3 подряд стоящих человека.

Известно, что на продажу i -му человеку из очереди одного билета кассир тратит A_i секунд, на продажу двух билетов — B_i секунд, трех билетов — C_i секунд. Напишите программу, которая подсчитает минимальное время, за которое могли быть обслужены все покупатели.

Обратите внимание, что билеты на группу объединившихся людей всегда покупает первый из них. Также никто в целях ускорения не покупает лишних билетов (то есть билетов, которые никому не нужны).

Формат входных данных

В первой строке входного файла находится число N ($1 \leq N \leq 5000$) - количество человек в очереди. В следующих N строках написаны по три натуральных числа A_i , B_i , C_i , каждое из этих чисел не превышает 3600. Люди в очереди нумеруются начиная от кассы.

Формат выходных данных

В выходной файл выведите одно число — минимальное время в секундах, за которое могли быть обслужены все покупатели.

Примеры

стандартный ввод	стандартный вывод
5	12
5 10 15	
2 10 15	
5 5 5	
20 20 1	
20 1 1	

Задача F. Транзисторы над Пекином

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Всемирно известный профессор В.В. Адимов продолжает свои разнообразные исследование устойчивости транзисторов. Теперь в голову ему пришла следующая задача: в доме N этажей, профессор хочет выяснить номер максимального этажа, падение с которого оставляет транзистор целым. Поскольку профессор исследует сферические транзисторы в вакууме, то можете считать что разбившись при падении с этажа f транзистор обязательно разобьется при падении с этажа $f + 1$. Дополнительно поставлено условие, что разрешено проведение не более чем K испытаний.

Эта задача была поручена именно вам, как самому успешному аспиранту профессора Адимова. Поскольку транзисторы нынче в цене, но наука все-таки дороже, то необходимо выяснить, какое минимальное количество транзисторов необходимо закупить, чтобы успешно провести эксперимент даже если вам будет катастрофически не везти.

Формат входных данных

В первой и единственной строке входного файла содержатся два целых положительных числа N и K не превосходящих 2000.

Формат выходных данных

Выведите единственное число - ответ на поставленную задачу. Если для данных N и K возможна ситуация, при которой мы не сможем получить ответ на вопрос даже имея неограниченный запас бесплатных транзисторов выведите -1 .

Примеры

	<code>stdin</code>	<code>stdout</code>
1	4 2	-1
2	4 3	2