

Задача А. Кузнечик

Имя входного файла: **grig.in**
Имя выходного файла: **grig.out**
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

У одного из преподавателей параллели С в комнате живёт кузнечик, который очень любит прыгать по клетчатой одномерной доске. Длина доски — N клеток. К его сожалению он умеет прыгать только на 1, 2, ..., k клеток вперёд.

Однажды преподавателям стало интересно, сколькими способами кузнечик может допрыгать из первой клетки до последней. Помогите им ответить на этот вопрос.

Формат входного файла

В первой и единственной строке входного файла записано два целых числа — N и k ($1 \leq N \leq 30, 1 \leq k \leq 10$).

Формат выходного файла

Выведите одно число — количество способов, которыми кузнечик может допрыгать из первой клетки до последней.

Пример

grig.in	grig.out
8 2	21

Задача В. Мутанты-2

Имя входного файла: **mutants2.in**
Имя выходного файла: **mutants2.out**
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Как вы помните, во Внесезонном Биологическом Колледже разводят милых разноцветных зверюшек. Но вдруг одна из зверюшек нашла выход из Колледжа и сбежала. По воле судьбы она попала в удивительный город Мутантославль. Вы не поверите, город разбит на улицы, на пересечении улиц находятся перекрестки.

Удивителен же Мутантославль тем, что ходить можно с перекрестка на перекресток только на восток или на юг, а также на каждом перекрестке берут штрафы. Наш мутант нашел карту города, она представляет собой клетчатый прямоугольник N на M , в котором на пересечении i -ой строки и j -ого столбца указан размер штрафа при попадании на этот перекресток.

Зверюшка находится на северо-западном углу города. Помогите ей дойти до юго-восточного угла Мутантославля, заплатив минимально возможный штраф.

Формат входного файла

В первой строке входного файла находятся два натуральных числа N и M ($1 \leq N, M \leq 1000$).

В последующих N строках содержатся по M чисел — карта города Мутантославль.

Формат выходного файла

В первой строчке выведите одно целое число — минимальный размер штрафа, который придется заплатить мутантику.

Во второй строчке выведите количество перекрестков на пути.

В следующих строчках выведите координаты перекрестков, через которые зверюшка пройдет. Гарантируется, что штраф не превысит 10^9 .

Пример

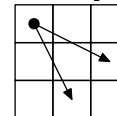
mutants2.in	mutants2.out
3 4	35
5 9 4 3	6
3 1 6 9	1 1
8 6 8 12	2 1
	2 2
	3 2
	3 3
	3 4

Задача С. Ход конём

Имя входного файла: **knight.in**
Имя выходного файла: **knight.out**
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 входного файла

В первой строке входного файла находятся два натуральных числа N и M ($1 \leq N, M \leq 50$).

Формат выходного файла

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

Пример

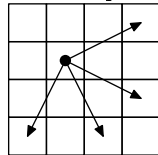
knight.in	knight.out
3 2	1
31 34	293930

Задача D. Ход конём - 2

Имя входного файла: knight2.in
Имя выходного файла: knight2.out
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 входного файла

В первой строке входного файла находятся два натуральных числа N и M ($1 \leq N, M \leq 50$).

Формат выходного файла

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

Пример

knight2.in	knight2.out
4 4	2
15 14	7884330

Задача E. Три единицы подряд

Имя входного файла: ones.in
Имя выходного файла: ones.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

По данному числу N определите количество последовательностей из нулей и единиц длины N , в которых никакие три единицы не стоят рядом.

Формат входного файла

Во входном файле написано натуральное число N , не превосходящее 35.

Формат выходного файла

Выведите количество искомых последовательностей. Гарантируется, что ответ не превосходит $2^{31} - 1$.

Пример

ones.in	ones.out
4	13

Задача F. Калькулятор

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

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

Формат выходного файла

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 .

Пример

calcul.in	calcul.out
1	0 1
5	3 1 2 4 5
962340	17 1 3 9 27 54 55 165 495 1485 4455 8910 17820 17821 53463 160389 160390 481170 962340

Задача G. Наибольшая возрастающая подпоследовательность

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

Дана последовательность, требуется найти длину её наибольшей возрастающей подпоследовательности.

Формат входного файла

В первой строке входных данных задано число N — длина последовательности ($1 \leq N \leq 1000$). Во второй строке задается сама последовательность (разделитель — пробел). Элементы последовательности — целые числа, не превосходящие 10000 по модулю.

Формат выходного файла

Требуется вывести длину наибольшей возрастающей подпоследовательности.

Пример

<code>sequence.in</code>	<code>sequence.out</code>
6 3 29 5 5 28 6	3

Задача H. Большой, белый, прямоугольный

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

В прямоугольной белой таблице клетки раскрашены в белый и черный цвета. Найти в ней прямоугольную область белого цвета, состоящую из наибольшего количества ячеек.

Формат входного файла

Во входном файле записана сначала высота N , а затем ширина M таблицы ($1 \leq N \leq 250, 1 \leq M \leq 250$), а затем записано N строк по M чисел в каждой строке, где 0 означает, что соответствующая клетка таблицы выкрашена в белый цвет, а 1 — что в черный.

Формат выходного файла

В выходной файл вывести одно число — количество клеток, содержащихся в наибольшем по площади белом прямоугольнике.

Пример

<code>bwhite.in</code>	<code>bwhite.out</code>
5 6 1 0 0 0 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0	9

Задача I. Наибольшая общая подпоследовательность

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

В первой строке входного файла записано число N — длина первой последовательности ($1 \leq N \leq 1000$).

Во второй строке записаны члены первой последовательности (через пробел) — целые числа, не превосходящие 10 000 по модулю.

В третьей строке записано число M — длина второй последовательности ($1 \leq M \leq 1000$).

В четвертой строке записаны члены второй последовательности (через пробел) — целые числа, не превосходящие 10 000 по модулю.

Формат выходного файла

В выходной файл требуется вывести единственное целое число: длину наибольшей общей подпоследовательности, или число 0, если такой не существует.

Пример

<code>lcs.in</code>	<code>lcs.out</code>
3 1 2 3 4 2 1 3 5	2

Задача J. Рюкзак

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

Найдите максимальный вес золота, который можно унести в рюкзаке вместительностью S , если есть N золотых слитков с заданными весами.

Формат входного файла

В первой строке входного файла записаны два числа — S и N ($1 \leq S \leq 10\,000$, $1 \leq N \leq 300$). Далее следует N неотрицательных целых чисел, не превосходящих $100\,000$ — веса слитков.

Формат выходного файла

Выведите искомый максимальный вес.

Пример

knapsack.in	knapsack.out
10 3 5 7 4	9