

Задача А. Цифры числа

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

Задано число N . Посчитайте количество N -значных чисел, у которых сумма цифр меньше произведения цифр. Так как ответ может быть очень большим, то надо вывести всего навсего остаток от деления этого ответа на $10^9 + 7$.

Формат входных данных

Единственная строка входного файла содержит одно целое число N ($1 \leq N \leq 50$)

Формат выходных данных

В выходной файл следует вывести единственное число – количество N -значных чисел, у которых сумма цифр меньше произведения цифр.

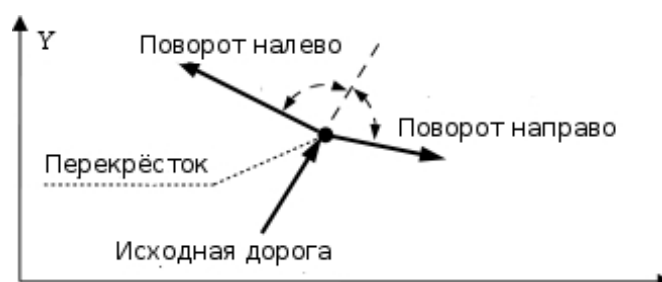
Примеры

<code>digits.in</code>	<code>digits.out</code>
2	63

Задача В. Налево

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

Как известно, схема городских дорог задаётся графом на плоскости. Вершины этого графа – перекрёстки – точки с целочисленными координатами на плоскости, а рёбра – дороги. Будем считать, что между двумя перекрёстками построено не более одной дороги и не существует дорог, соединяющих перекресток с самим собой. Ваша задача – найти такой путь от перекрёстка A до перекрёстка B , чтобы на всех перекрёстках поворачивать только налево. Поворотами налево считаются все повороты на угол не более 180 градусов против часовой стрелки. Путь с перекрёстка A можно начинать в любую сторону.



Формат входных данных

Первая строка входного файла содержит N и M – количество перекрёстков и количество дорог ($1 \leq N \leq 10000, 1 \leq M \leq 10000$). Вторая строка содержит числа a и b – номера перекрёстков A и B соответственно. Каждая из следующих N строк содержит пару целых чисел через пробел X_i и Y_i – координаты перекрёстка i . Последние M строк содержат $2M$ чисел, каждая пара чисел – номера перекрёстков соединенных дорогой. Все координаты по модулю не превосходят 10000. На одном перекрестке не могут заканчиваться более 20 дорог.

Формат выходных данных

Если, сохраняя вышеуказанные правила невозможно добраться от перекрёстка A до перекрёстка B , то нужно вывести -1 . Иначе нужно вывести количество дорог на этом пути, а затем номера соответствующих дорог в порядке объезда. Если по одной и той же дороге необходимо проехать 2 раза подряд номер этой дороги нужно вывести только один раз. Если есть несколько вариантов ответа, вывести любой.

Примеры

left.in	left.out
2 1	1 1
1 2	
0 0	
1 1	
1 2	

Задача C. QBIT

Имя входного файла: qbit.in
Имя выходного файла: qbit.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 Mb

Задана строка, состоящая только из заглавных латинских букв. За одно действие из этой строки можно удалить букву или несколько стоящих рядом букв. Напишите программу, которая находит наименьшее количество действий, за которые можно из заданной строки получить слово QBIT.

Формат входных данных

Во входном файле содержится единственная строка, состоящая из не менее 4 и не более 256 заглавных букв латинского алфавита.

Формат выходных данных

Если слово QBIT нельзя получить, то в выходной файл следует вывести -1, иначе – единственное число – наименьшее количество действий, за которые можно из заданной строки получить слово QBIT.

Примеры

qbit.in	qbit.out
QUBBIIT	2

Задача D. Флажки

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

На день рождения Ульяма его родители решили украсить его комнату квадратными флажками. Каждый флажок шьется из девяти одинаковых по размеру квадратных лоскутков, причём каждый из лоскутков может быть одного из K цветов. Напишите программу, которая посчитает количество различных с точностью до поворотов и переворотов флажков, которые можно подарить на день рождения Уильяма.

Формат входных данных

В единственной строке входного файла записано одно число K ($1 \leq K \leq 100$)

Формат выходных данных

В выходной файл выведите единственное число – количество различных с точностью до поворотов и переворотов флажков, которые можно подарить на день рождения Уильяма.

Примеры

	<code>flags.in</code>	<code>flags.out</code>
2	2	102
3	3	2862

Задача Е. Прямые на плоскости

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

Как известно, несколько несовпадающих прямых разбивают плоскость на несколько частей, причём некоторые из этих частей имеют бесконечную площадь, а некоторые – конечную (это выпуклые многоугольники). Ваша задача – посчитать количество частей с конечной площадью.

Формат входных данных

В первой строке входного файла записано целое число N ($1 \leq N \leq 50$) – количество прямых на плоскости. Каждая из следующих N строк описывает одну из прямых, прямая задаётся координатами двух точек. Все координаты не превосходят 10000 по абсолютной величине.

Формат выходных данных

В выходной файл выведите одно число – количество частей с конечной площадью, которые получаются при разбиении плоскости заданным набором прямых.

Примеры

<code>lines.in</code>	<code>lines.out</code>
3 0 0 0 10 0 0 10 0 0 5 5 0	1

Задача F. Максимальный результат

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

Даны N чисел, надо разбить их на пары и оставить некоторые без пары. Далее сложим произведения в парах с элементами без пары. Ваша задача – определить максимальный возможный результат. Например, пусть есть 6 чисел: 4, 1, 8, 3, 2, 1. Поступим с ними так как указано выше, например (3, 8), (4, 1), (2, 1); результат – $3 * 8 + 4 * 1 + 2 * 1 = 30$. А если так: (4, 8), (3, 2), (1), (1), то результат – $4 * 8 + 3 * 2 + 1 + 1 = 40$.

Формат входных данных

Первая строка входного файла содержит одно целое число N ($1 \leq N \leq 50$). Далее следуют N чисел. Все числа не превосходят 1000 по абсолютной величине.

Формат выходных данных

В выходной файл следует вывести единственное число – максимальный возможный результат.

Примеры

<code>max.in</code>	<code>max.out</code>
3 1 2 3	7
4 1 1 1 1	4

Задача G. Кратчайший путь с телепортами

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

На прямоугольной доске N на M клеточек, часть клеток – телепорты. Верхний левый угол – клеточка с координатами $(1, 1)$. В клеточке с координатами (x_1, y_1) находится «сказочный герой», который желает как можно быстрее попасть в клеточку с координатами (x_2, y_2) . За одну секунду герой может либо переместиться в соседнюю клетку, т.е. клетку, имеющую общую сторону с той, в которой он находится, либо, если в текущей клетке есть телепорт, мгновенно (т.е. за 0 секунд) телепортироваться в любой другой телепорт.

Напишите программу, которая найдёт кратчайший по времени путь из клеточки с координатами (x_1, y_1) в клеточку с координатами (x_2, y_2) .

Формат входных данных

В первой строке входного файла записаны два числа N ($2 \leq N \leq 1000$) – количество клеточек по вертикали, M ($2 \leq M \leq 1000$) – количество клеточек по горизонтали. В следующих N строках, в каждой по M символов записана доска. Буква “T” – обозначает телепорт, “.” – обозначает обычную клетку. В следующей строке записаны числа x_1, y_1, x_2, y_2 .

Формат выходных данных

В выходной файл выведите единственное число – минимальное время, за которое «сказочный герой» может попасть из клеточки (x_1, y_1) в клеточку (x_2, y_2) .

Примеры

<code>teleport.in</code>	<code>teleport.out</code>
5 5T...T. 1 1 5 5	4

Задача Н. Остатки от деления

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

Задано число A . Напишите программу, которая находит число N , дающее остаток 1 от деления на 2, 2 от деления на 3, 3 от деления на 4, ..., $A - 1$ – от деления на A .

Формат входных данных

Во входном файле записано единственное число A ($2 \leq A \leq 1000$).

Формат выходных данных

В выходной файл выведите единственное число N , не превосходящее 10^{10000} и удовлетворяющее условию задачи.

Примеры

<code>modulo.in</code>	<code>modulo.out</code>
4	23

Задача I. Золотые слитки

Имя входного файла: gold.in
Имя выходного файла: gold.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 Mb

Дано N золотых слитков. Требуется распилить не более одного из них на две части (не обязательно равные, но с целой массой), после чего разделить слитки на две кучи равной массы.

Формат входных данных

В первой строке вводится одно натуральное число N , не превосходящее 100.

Во второй строке через пробел вводятся N натуральных чисел, не превосходящих 100 — массы имеющихся слитков.

Формат выходных данных

Выведите массы слитков, которые вошли в первую кучку (включая массу части распиленного слитка).

Если решений несколько, выведите любое из них.

Если решений нет, выведите фразу NO SOLUTION (заглавными буквами).

Выводить массы можно в произвольном порядке, но масса части распиленного слитка (если таковой имеется) должна быть последней.

Примеры

gold.in	gold.out
3 5 5 5	NO SOLUTION
4 1 2 3 8	2 3 1 1
4 1 2 3 8	7

Задача J. Сказочные монеты

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

Однажды наш «сказочный герой», отправляясь в путешествие за границу, решил взять с собой такое количество «сказочных монет», чтобы ими можно было дать таможеннику взятку на любую сумму в диапазоне от 1 до некоторого числа X . Всего есть несколько типов монет и у каждого типа монет своя стоимость. Понятное дело, что герой хочет добиться того, чтобы общее количество монет, которые он возьмёт с собой, было минимальным. Напишите программу, которая определит, можно ли вообще собрать набор монет, удовлетворяющий условию. И, если можно, то определите, какое минимальное количество «сказочных монет» должно быть в таком наборе.

Формат входных данных

В первой строке входного файла записано число X ($1 \leq X \leq 1000$). Во второй строке записаны несколько чисел (не более, чем 50) – стоимости каждого типа монет.

Формат выходных данных

Если собрать набор монет, удовлетворяющих условию не удастся, то в выходной файл выведите -1 . Иначе в выходной файл выведите единственное число – минимальное количество сказочных монет, которое должно быть в таком наборе.

Примеры

<code>money.in</code>	<code>money.out</code>
1000 1	1000
1000 2 1	501