

## Задача А. Кузнечик

Имя входного файла: `grig.in`  
Имя выходного файла: `grig.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

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  $1, 2, \dots, k$  клеток вперёд.

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

### Формат входных данных

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 выходных данных

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

### Пример

<code>grig.in</code>	<code>grig.out</code>
8 2	21

## Задача В. Лестница

Имя входного файла: `ladder.in`  
Имя выходного файла: `ladder.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

У лестницы  $n$  ступенек, пронумерованных числами  $1, 2, \dots, n$  снизу вверх. На каждой ступеньке написано число. Начиная с подножия лестницы (его можно считать ступенькой с номером 0), требуется взобраться на самый верх (ступеньку с номером  $n$ ). За один шаг можно подниматься на одну или на две ступеньки. После подъёма числа, записанные на посещённых ступеньках, складываются. Нужно подняться по лестнице так, чтобы сумма этих чисел была как можно больше.

### Формат входных данных

В первой строке входного файла записано целое число  $n$  ( $1 \leq n \leq 100$ ). Во второй строке заданы целые числа  $a_1, a_2, \dots, a_n$  через пробел ( $-10\,000 \leq a_i \leq 10\,000$ ) — это числа, записанные на ступеньках.

### Формат выходных данных

В первой строке выходного файла выведите одно число — максимальную сумму, которую можно получить, поднявшись по данной лестнице.

### Пример

<code>ladder.in</code>	<code>ladder.out</code>
2 1 2	3

## Задача С. Черепашка

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

В левом верхнем углу прямоугольной таблицы размером  $N \times M$  находится черепашка. На каждой клетке этой таблицы разлито некоторое количество кислоты. Черепашка может перемещаться вправо или вниз, при этом маршрут черепашки заканчивается в правом нижнем углу таблицы.

Каждый миллилитр кислоты наносит черепашке некоторое количество урона. Найдите наименьшее возможное значение урона, который получит черепашка после прогулки по таблице.

### Формат входных данных

В первой строке входных данных записаны два натуральных числа  $N$  и  $M$ , не превосходящих 1000 — размеры таблицы. Далее идет  $N$  строк, каждая из которых содержит  $M$  натуральных чисел, разделенных пробелами — описание таблицы с указанием для каждой клетки содержания кислоты на ней (в миллилитрах). Гарантируется, что содержание кислоты в клетке не превышает 250 миллилитров.

### Формат выходных данных

Программа должна вывести единственное число: минимальную возможную стоимость маршрута черепашки.

### Примеры

стандартный ввод	стандартный вывод
3 4 5 9 4 3 3 1 6 9 8 6 8 12	35
1 1 1	1

## Задача D. Рюкзак

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

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

### Формат входных данных

В первой строке задано два целых числа  $S$  и  $n$  ( $1 \leq n \leq 300$ ;  $1 \leq S \leq 10\,000$ ).

Далее следует  $n$  неотрицательных целых чисел, не превосходящих 100 000 — веса слитков.

### Формат выходных данных

Выведите искомый максимальный вес.

### Примеры

стандартный ввод	стандартный вывод
10 3 1 4 8	9
20 4 5 7 12 18	19

## Задача Е. Рюкзак

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дано  $n$  предметов массой  $m_1, \dots, m_n$  и стоимостью  $c_1, \dots, c_n$  соответственно.

Ими наполняют рюкзак, который выдерживает вес не более  $m$ . Определите набор предметов, который можно унести в рюкзаке, имеющий наибольшую стоимость.

### Формат входных данных

В первой строке вводится натуральное число  $n$ , не превышающее 1000 и натуральное число  $m$ , не превышающее 10000.

Во второй строке вводятся  $n$  натуральных чисел  $m_i$ , не превышающих 100.

Во третьей строке вводятся  $n$  натуральных чисел  $c_i$ , не превышающих 100.

### Формат выходных данных

В первой строке выведите количество предметов, которые нужно взять. Во второй строке выведите номера предметов (числа от 1 до  $n$ ), которые войдут в рюкзак наибольшей стоимости.

### Пример

стандартный ввод	стандартный вывод
4 6 2 4 1 2 7 2 5 1	3 1 3 4

## Задача F. Зайчик

Имя входного файла: `lepus.in`  
Имя выходного файла: `lepus.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Зайчик прыгает по прямой просеке, для удобства разделённой на  $n$  клеток. Клетки пронумерованы по порядку натуральными числами от 1 до  $n$ . Некоторые клетки заболочены: если зайчик прыгнет на такую клетку, ему несдобровать. Некоторые другие клетки просеки поросли вкусной зелёной травой: прыгнув на такую клетку, зайчик сможет отдохнуть и подкрепиться.

Зайчик начинает свой путь из клетки с номером 1 и хочет попасть в клетку с номером  $n$ , по пути ни разу не провалившись в болото и скушав как можно больше вкусной зелёной травы. Конструктивные особенности зайчика таковы, что из клетки с номером  $k$  он может прыгнуть лишь в клетки с номерами  $k + 1$ ,  $k + 3$  и  $k + 5$ .

Выясните, какое максимальное количество клеток с травой сможет посетить зайчик на своём пути.

### Формат входных данных

В первой строке входного файла задано число  $n$  — количество клеток ( $2 \leq n \leq 1000$ ). Вторая строка состоит из  $n$  символов;  $i$ -ый символ соответствует  $i$ -ой клетке просеки. Символ 'w' обозначает болото, символ '.' — зелёную траву, а символ '.' соответствует клетке без каких-либо особенностей. Гарантируется, что первая и последняя клетки не содержат болот и травы.

### Формат выходных данных

В первой строке выходного файла выведите одно число — максимальное количество клеток с травой, которые зайчик сможет посетить на своём пути. Если зайчику не удастся оказаться в клетке с номером  $n$ , выведите «-1».

### Примеры

lepus.in	lepus.out
4 ..	2
5 .w".	0

## Задача G. Наибольшая возрастающая подпоследовательность

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дана последовательность, требуется найти её наибольшую возрастающую подпоследовательность.

### Формат входных данных

В первой строке входных данных задано целое число  $n$  — длина последовательности ( $1 \leq n \leq 5000$ ). Во второй строке задается сама последовательность. Числа разделяются пробелом. Элементы последовательности — целые числа, не превосходящие  $10^9$  по модулю.

### Формат выходных данных

В первой строке выведите длину наибольшей возрастающей подпоследовательности, а во второй строке выведите через пробел саму наибольшую возрастающую подпоследовательность данной последовательности. Если ответов несколько — выведите любой.

### Пример

стандартный ввод	стандартный вывод
6 3 29 5 5 28 6	3 3 5 28

## Задача Н. Числа фиксированной суммы

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Сколько существует  $K$ -значных чисел с суммой цифр равной  $S$ ? Числа берутся в десятичной системе счисления. Ведущие нули допустимы.

### Формат входных данных

$K$  ( $1 \leq K \leq 1000$ ) и  $S$  ( $0 \leq S \leq 9 \cdot K$ ).

### Формат выходных данных

Количество таких чисел, посчитанное по модулю  $10^9 + 7$ .

### Примеры

стандартный ввод	стандартный вывод
3 10	63



## Задача I. Плавные числа

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

Назовём натуральное число плавным, если разность любых двух его соседних цифр не превосходит по модулю единицы. Определите количество  $N$ -значных плавных чисел. Запись числа не может начинаться с цифры 0.

### Формат входных данных

Программа получает на вход одно целое число  $N$  ( $1 \leq N \leq 20$ ).

### Формат выходных данных

Программа должна вывести одно целое число — искомое количество плавных чисел.

### Пример

<code>numbers.in</code>	<code>numbers.out</code>
2	26

## Задача J. Наибольшая общая подпоследовательность

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Даны две последовательности, требуется найти и вывести их наибольшую общую подпоследовательность.

### Формат входных данных

В первой строке входных данных содержится целое число  $n$  — длина первой последовательности ( $1 \leq n \leq 2000$ ). Во второй строке заданы члены первой последовательности (через пробел) — целые числа, не превосходящие  $10^9$  по модулю. В третьей строке записано целое число  $m$  — длина второй последовательности ( $1 \leq m \leq 2000$ ). В четвертой строке задаются члены второй последовательности (через пробел) — целые числа, не превосходящие  $10^9$  по модулю.

### Формат выходных данных

В первой строке выведите длину наибольшей общей подпоследовательности, а во второй строке выведите через пробел саму наибольшую общую подпоследовательность данных последовательностей. Если ответов несколько — выведите любой.

### Пример

стандартный ввод	стандартный вывод
3 1 2 3 4 2 3 1 5	2 2 3

## Задача К. Ход конем

Имя входного файла: `phones.in`  
Имя выходного файла: `phones.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Шахматная ассоциация решила оснастить всех своих сотрудников такими телефонными номерами, которые бы набирались на кнопочном телефоне ходом коня. Например, ходом коня набирается телефон 340-49-27. При этом телефонный номер не может начинаться ни с цифры 0, ни с цифры 8.

1	2	3
4	5	6
7	8	9
	0	

Напишите программу, определяющую количество телефонных номеров длины  $N$ , набираемых ходом коня. Поскольку таких номеров может быть очень много, выведите ответ по модулю  $10^9$ .

### Формат входных данных

Во входном файле записано целое число  $N$  ( $1 \leq N \leq 100$ ).

### Формат выходных данных

Выведите в выходной файл искомое количество телефонных номеров по модулю  $10^9$ .

### Примеры

<code>phones.in</code>	<code>phones.out</code>
1	8
2	16

## Задача L. Кролик учит геометрию.

Имя входного файла: rabbits.in  
Имя выходного файла: rabbits.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Кролики — очень любопытны. Они любят изучать геометрию, бегая по грядкам. Наш кролик как раз из таких. Сегодня он решил изучить новую фигуру — квадрат.

Кролик бегает по грядке — клеточному полю  $N \times M$  клеток. В некоторых из них посеяны морковки, в некоторых нет.

Помогите Кролику найти сторону квадрата наибольшей площади, заполненного морковками полностью.

### Формат входных данных

В первой строке даны два натуральных числа  $N$  и  $M$  ( $1 \leq N, M \leq 1000$ ). Далее в  $N$  строках расположено по  $M$  чисел, разделенных пробелами (число равно 0, если в клетке нет морковки или 1, если есть).

### Формат выходных данных

Выведите одно число — сторону максимального квадрата, заполненного морковками.

### Пример

rabbits.in	rabbits.out
4 5 0 0 0 1 0 0 1 1 1 0 0 0 1 1 0 1 0 1 0 0	2

## Задача М. Принятие решений

Имя входного файла: `decisions.in`  
Имя выходного файла: `decisions.out`  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 512 мегабайт

Денис готовится к поездке с классом. У Дениса есть несколько предметов, которые он планирует взять с собой в дорогу. Он разложил их в одну линию и пронумеровал от 1 до  $n$ . Для каждого предмета Денис знает его вес  $w_i$ . Рюкзак, которые собирается взять Денис, имеет вместительность  $c$ , это значит что в него можно положить предметы суммарного веса не более  $c$ .

К сожалению, Денис не очень хорош в принятии решений. Он не может решить, какие предметы будут более полезными в поездке. Также он не любит носить с собой много, так что возможно, он не будет заполнять свой рюкзак полностью. Чтобы помочь себе выбрать, Денис решил установить дополнительные ограничения на предметы, которые он выберет.

Теперь Денис генерирует разные планы ограничений. Каждый план это тройка чисел  $(l_j, r_j, t_j)$ , где  $1 \leq l_j \leq r_j \leq n$  и  $0 \leq t_j \leq c$ . Чтобы удовлетворить ограничения, Денис должен выбрать несколько предметов с номерами между  $l_j$  и  $r_j$ , включительно, с суммарным весом равным  $t_j$ . Для каждого плана Денис хочет сразу же узнать может он его удовлетворить или нет. Помогите ему!

### Формат входных данных

Входные данные содержат несколько тестов.

Каждый тест начинается с целого числа  $n$  ( $1 \leq n \leq 2000$ ). Вторая строка каждого теста содержит  $n$  целых чисел  $w_i$  ( $1 \leq w_i \leq 100$ ). Третья строка содержит вместительность рюкзака  $c$  ( $1 \leq c \leq 1000$ ). Четвертая строка содержит  $q$  — число планов, которые Денис собирается предложить ( $1 \leq q \leq 300\,000$ ). Следующие  $q$  строк описывают планы.

План с номером  $j$  описан тремя целыми числами:  $a_j, b_j, c_j$  ( $0 \leq a_j < n$ ;  $1 \leq b_j \leq n$ ;  $0 \leq c_j \leq c$ ). Пусть в этом тесте уже было  $k$  планов до  $j$ -го, которые могли быть удовлетворены. Используйте следующее уравнение, чтобы получить  $j$ -й план:  $l_j = (a_j + k) \bmod (n - b_j + 1) + 1$ ,  $r_j = l_j + b_j - 1$ , and  $t_j = (c_j + k) \bmod (c + 1)$ .

После последнего теста следует одна строка, состоящая из единственного 0. Сумма всех  $n$  по всем тестам не превосходит 2000. Сумма всех  $q$  по всем тестам не превосходит 300 000.

### Формат выходных данных

Для каждого теста выведите строку из символов 'Y' and 'N'. Для каждого плана выведите 'Y', если он может быть удовлетворим, или 'N' иначе.

### Пример

decisions.in	decisions.out
5 2 3 4 5 9 10 5 0 3 5 0 2 4 0 3 4 0 4 6 2 3 5 0	YNYYN