

Содержание

Must have	2
Задача 9А. Калькулятор [0.2 sec, 256 mb]	2
Задача 9В. Функция [0.2 sec, 256 mb]	3
Задача 9С. Рюкзак [0.1 sec, 256 mb]	4
Обязательные задачи	5
Задача 9D. Три последовательности [0.1 sec, 256 mb]	5
Задача 9Е. Три последовательности [0.1 sec, 256 mb]	6
Задача 9F. Редакционное расстояние [0.1 sec, 256 mb]	7
Задача 9G. Гвоздики [0.1 sec, 256 mb]	8
Задача 9H. Поле [0.2 sec, 256 mb]	9
Задача 9I. Невозрастающая подпоследовательность [0.5 sec, 256 mb]	10
Дополнительные задачи	11
Задача 9J. Беру? Нет, не беру! [0.3 sec, 256 mb]	11
Задача 9K. Различные подпоследовательности [0.2 sec, 256 mb]	12
Задача 9L. Психотренинг [0.1 sec, 256 mb]	13
Задача 9M. Капитан Оптимальность [4.5 sec, 256 mb]	14

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же stdin), вывести ответ нужно в **стандартный поток вывода** (он же stdout).

В некоторых задачах большой ввод и вывод. Пользуйтесь **быстрым вводом-выводом**.

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) **переопределение стандартного аллокатора** ускорит вашу программу.

Обратите внимание на GNU C++ компиляторы с суффиксом inc, они позволяют пользоваться **дополнительной библиотекой**. Под ними можно сдать **вот это**.

Must have

Задача 9А. Калькулятор [0.2 sec, 256 mb]

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 .

Примеры

stdin	stdout
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

Замечание

Разобрана на лекции.

Задача 9В. Функция [0.2 сек, 256 mb]

Вычислите функцию:
$$f(n) = \begin{cases} 1 & \text{если } n \leq 2 \\ f(\lfloor 6 * n/7 \rfloor) + f(\lfloor 2 * n/3 \rfloor) & \text{если } n \bmod 2 = 1 \\ f(n-1) + f(n-3) & \text{если } n \bmod 2 = 0 \end{cases}$$

Формат входных данных

Входные данные содержат натуральное число n ($1 \leq n \leq 10^{12}$).

Формат выходных данных

Выведите значение функции по модулю 2^{32} .

Пример

stdin	stdout
7	10

Замечание

Будьте достаточно ленивы.

Задача 9С. Рюкзак [0.1 sec, 256 mb]

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 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 выходных данных

Выведите искомый максимальный вес.

Примеры

stdin	stdout
10 3 1 4 8	9
20 4 5 7 12 18	19

Замечание

Разобрана на лекции.

Обязательные задачи

Задача 9D. Три последовательности [0.1 sec, 256 mb]

Даны три последовательности целых чисел. Ваша задача — найти **длину** их наибольшей общей подпоследовательности.

Формат входных данных

Входной файл содержит описание трех последовательностей. Каждая последовательность задается двумя строками. Первая строка содержит длину последовательности n ($1 \leq n \leq 100$), а вторая — ее элементы (32-х битные целые числа).

Формат выходных данных

Выведите длину наибольшей общей подпоследовательности.

Примеры

stdin	stdout
3 1 2 3 3 2 1 3 3 1 3 5	2
3 1 2 3 3 4 5 6 3 1 3 5	0

Замечание

НОП двух и НОП трёх – конечно, очень похожие задачи.
НОП двух разобрана на лекции.

Задача 9E. Три последовательности [0.1 sec, 256 mb]

Даны три последовательности целых чисел. Ваша задача — найти их наибольшую общую подпоследовательность.

Формат входных данных

Входной файл содержит описание трех последовательностей. Каждая последовательность задается двумя строчками. Первая строка содержит длину последовательности n ($1 \leq n \leq 100$), а вторая — ее элементы (32-х битные целые числа).

Формат выходных данных

Первая строка выходного файла должна содержать длину максимальной общей подпоследовательности. Саму подпоследовательность необходимо вывести во второй строке. Если таких строк несколько, можно вывести любую из них.

Примеры

stdin	stdout
3 1 2 3 3 2 1 3 3 1 3 5	2 1 3
3 1 2 3 3 4 5 6 3 1 3 5	0

Замечание

НОП двух и НОП трёх – конечно, очень похожие задачи.
Нужно будет немного попотеть с восстановлением ответа.

Задача 9F. Редакционное расстояние [0.1 sec, 256 mb]

В информатике *редакционным расстоянием* между двумя строками называется минимальное количество добавлений, удалений и замен символов, при помощи которых можно из одной строки получить другую. К примеру, редакционное расстояние между строками «ab» и «ab» равно нулю, так как строки равны между собой безо всяких изменений; расстояние между строками «short» и «ports» равно трём: в слове «short» нужно удалить из начала букву 's', заменить 'h' на 'p' и добавить в конец букву 's'. Редакционное расстояние также называют *расстоянием Левенштейна*.

Найдите редакционное расстояние между двумя заданными строками.

Формат входных данных

В первой строчке входного файла задана одна строка, во второй — другая.

Длины строк от 1 до 100.

Формат выходных данных

В выходной файл выведите единственное число — редакционное расстояние между двумя заданными строками.

Примеры

stdin	stdout
ab ab	0
short ports	3

Замечание

Разобрана на практике.

Задача 9G. Гвоздики [0.1 sec, 256 mb]

На прямой дощечке вбиты гвоздики. Любые два гвоздика можно соединить ниточкой. Требуется соединить какие-то пары гвоздиков ниточками так, чтобы к каждому гвоздику была привязана хотя бы одна ниточка, а суммарная длина всех ниточек была минимальна.

Формат входных данных

В первой строке входного файла записано число N — количество гвоздиков ($2 \leq N \leq 100$). В следующей строке записано N чисел — координаты всех гвоздиков (неотрицательные целые числа, не превосходящие 10 000).

Формат выходных данных

В выходной файл нужно вывести единственное число — минимальную суммарную длину всех ниточек.

Пример

stdin	stdout
5	6
4 10 0 12 2	

Замечание

Это задача на динамику. Думайте «идём слева направо...»

Задача 9Н. Поле [0.2 sec, 256 mb]

Отряду нужно пересечь прямоугольное поле размера $m \times n$ квадратов, двигаясь из левого верхнего угла в правый нижний и перемещаясь между соседними квадратами только в двух направлениях — вправо и вниз. Поле не очень ровное, но у отряда есть карта, на которой отмечена высота каждого квадрата. Опасность перехода с квадрата высоты h_1 на соседний квадрат высоты h_2 оценивается числом $|h_2 - h_1|$; опасность всех переходов в пути суммируется. Выясните, какова минимальная опасность пути из квадрата $(1, 1)$ в квадрат (m, n) .

Формат входных данных

В первой строке входного файла заданы два числа m и n через пробел ($1 \leq m, n \leq 100$). В следующих n строках записано по m чисел в каждой; i -ое число j -ой из этих строк соответствует высоте квадрата (i, j) . Все высоты — целые числа в диапазоне от 1 до 100, включительно.

Формат выходных данных

Выведите в выходной файл одно число — минимальную опасность пути из квадрата $(1, 1)$ в квадрат (m, n) .

Примеры

stdin	stdout
2 2 1 1 1 1	0
4 2 1 2 3 5 3 8 4 7	6
2 3 1 2 2 3 3 1	4

Задача 9I. Невозрастающая подпоследовательность [0.5 sec, 256 mb]

Вам требуется написать программу, которая по заданной последовательности находит максимальную невозрастающую её подпоследовательность (т.е такую последовательность чисел $a_{i_1}, a_{i_2}, \dots, a_{i_k}$ ($i_1 < i_2 < \dots < i_k$), что $a_{i_1} \geq a_{i_2} \geq \dots \geq a_{i_k}$ и не существует последовательности с теми же свойствами длиной $k + 1$).

Формат входных данных

В первой строке задано число n — количество элементов последовательности ($1 \leq n \leq 239\,017$). В последующих строках идут сами числа последовательности a_i , отделенные друг от друга произвольным количеством пробелов и переводов строки (все числа не превосходят по модулю $2^{31} - 2$).

Формат выходных данных

Вам необходимо выдать в первой строке выходного файла число k — длину максимальной невозрастающей подпоследовательности. В последующих строках должны быть выведены (по одному числу в каждой строке) все номера элементов исходной последовательности i_j , образующих искомую подпоследовательность. Номера выводятся в порядке возрастания. Если оптимальных решений несколько, разрешается выводить любое.

Пример

stdin	stdout
5	3
5 8	2
10 4 1	4
	5

Замечание

Закрепляем пройденное на лекции.

Всем рекомендуем написать, не подглядывая в конспект ;-)

Дополнительные задачи

Задача 9J. Беру? Нет, не беру! [0.3 сек, 256 mb]

Саша и Петя играют в весёлую игру. Петя рисует на листе бумаги таблицу $m \times n$ и заполняет её целыми числами. После этого Саша ставит свою фишку на клетку $(1, 1)$ и может каждым своим ходом двигать её вправо или вниз. Кроме того, если Сашина фишка находится на числе большем, чем все числа, которые он брал до этого, Саша может сказать «беру», и тогда Петя записывает на бумажку то число, на котором стоит Сашина фишка (если Саша не брал ранее ни одного числа, то он может сказать «беру» на любом числе). Игра заканчивается, когда Сашина фишка больше не может ходить, и количество очков, которые набрал Саша, зависит от того, сколько чисел он взял. Помогите Саше «взять» как можно больше чисел!

Формат входных данных

Входной файл состоит из одного или нескольких наборов входных данных (не более 5). Набор входных данных начинается строкой, в которой записаны числа m и n ($1 \leq m, n \leq 100$). Далее следуют n строк по m чисел в каждой — таблица, которую нарисовал Петя. Все числа не превосходят по модулю 10000.

Файл завершается набором с $m = n = 0$.

Формат выходных данных

Для каждого из наборов входных данных выведите в первой строке максимальное количество чисел M , которые может взять Саша. Во второй строке выведите один из возможных вариантов Сашиних действий — строку из символов R, D и T, где R означает ход вправо ($x \leftarrow x + 1$), D обозначает ход вниз ($y \leftarrow y + 1$) и T обозначает взятие числа, на котором в данный момент стоит фишка.

Разделяйте вывод для разных наборов входных данных одной пустой строкой.

Пример

stdin	stdout
2 2	1
1 1	TRD
1 1	
2 3	3
1 2	TRTDDT
1 1	
1 3	
0 0	

Замечание

Задача на силу мозга и рук.

Не пишите, пожалуйста, деревья. Особенно многомерные. Лучше используйте мозг.

Задача 9К. Различные подпоследовательности [0.2 sec, 256 mb]

Дана последовательность целых чисел длины n . Нужно найти количество различных непустых подпоследовательностей.

Формат входных данных

На первой строке n ($1 \leq n \leq 300\,000$). На второй строке n целых чисел от 0 до 300 000.

Формат выходных данных

Количество различных подпоследовательностей по модулю $10^9 + 7$.

Пример

stdin	stdout
3 1 1 2	5
35 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5 6 7 8 9 10 1 2 3 4 5	941167856

Замечание

Красивое простое решение.

Задача 9L. Психотренинг [0.1 sec, 256 mb]

На очередном психологическом тренинге n участников сборов играют в занимательную игру. Участники игры рассаживаются по кругу и получают номера от 1 до n против часовой стрелки. После этого главный психолог отсчитывает против часовой стрелки k -го участника игры, начиная с первого. Этот участник выходит из круга и может идти на ужин. А остальные продолжают участие в тренинге. Главный психолог отсчитывает еще k участников, начиная со следующего после выбывшего. Участник, который оказался k -ым, тоже покидает тренинг, и так далее.

Участники сборов решили сесть в круг таким образом, чтобы один вредный тип пошел ужинать последним. Для этого они хотят установить, какой номер он должен для этого получить. Помогите им.

Формат входных данных

Входной файл содержит два целых числа: n и k ($1 \leq n \leq 10^{18}$, $1 \leq k \leq 1000$).

Формат выходных данных

Выведите в выходной файл одно число — номер участника, который пойдет на ужин последним.

Пример

stdin	stdout
5 3	4

Замечание

Привет с древних сборов к межнару ;-)

Задача 9M. Капитан Оптимальность [4.5 сек, 256 mb]

Имеется калькулятор, который умеет выполнять с текущим числом на экране следующие операции:

- Умножить текущее число на A .
- Умножить текущее число на B .
- Прибавить к текущему числу единицу.
- Прибавить к текущему числу число C .

Здесь A , B и C — некоторые целочисленные константы. Определите, какое наименьшее количество операций требуется, чтобы получить из числа 1 число n .

Формат входных данных

Во вводе задано несколько строк. Каждая строка содержит очередной тестовый случай, задаваемый четырьмя целыми числами n , A , B и C ($1 \leq n \leq 10^{19}$, $2 \leq A, B \leq 10$, $1 \leq C \leq 10$). Входные данные содержат не более 100 тестов.

Формат выходных данных

Для каждого тестового случая выведите $k + 1$, где k — количество сделанных операций. За ним выведите все $k + 1$ промежуточных чисел, включая 1 и n . Если последовательностей из k операций несколько, можно вывести любую из них.

Пример

stdin	stdout
1 7 8 9	1 1
5 2 3 9	4 1 2 4 5
576 8 9 7	4 1 8 64 576
100000 2 3 1	19 1 2 6 18 19 38 76 77 231 693 694 2082 2083 6249 6250 12500 25000 50000 100000

Подсказка по решению

Ленивость и аккуратность.