

Содержание

Задачи здорового человека	2
Задача 6А. Разложение на кубы [0.2 sec, 256 mb]	2
Задача 6В. Калькулятор [0.2 sec, 256 mb]	3
Задача 6С. Наибольшая общая подпоследовательность [0.2 sec, 256 mb]	4
Задача 6D. Поле [0.2 sec, 256 mb]	5
Задача 6Е. Редакционное расстояние [0.1 sec, 256 mb]	6
Задача 6F. Рюкзак [0.1 sec, 256 mb]	7
Для искателей острых ощущений	8
Задача 6G. Три последовательности [0.1 sec, 256 mb]	8
Задача 6H. Гвоздики [0.1 sec, 256 mb]	9

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**.

Задачи здорового человека

Задача 6А. Разложение на кубы [0.2 sec, 256 mb]

Дано целое число от n , представить его в виде суммы минимального числа кубов.

Формат входных данных

В этой задаче мультитест. Каждая строка входных данных содержит одно целое n от 1 до 50 000. Количество n в одном тесте от 1 до 10 000.

Формат выходных данных

Для каждого n на отдельной строке минимальное количество кубов в разложении.

Примеры

stdin	stdout
7	7
8	1
10	3
43	3

Замечание

$$7 = 1^3 + 1^3 + 1^3 + 1^3 + 1^3 + 1^3 + 1^3$$

$$8 = 2^3$$

$$10 = 2^3 + 1^3 + 1^3$$

$$43 = 2^3 + 2^3 + 3^3$$

Подсказка по решению

Разобрана на практике.

Задача 6В. 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 6С. Наибольшая общая подпоследовательность [0.2 sec, 256 mb]

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 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 выходных данных

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

Примеры

stdin	stdout
3	2
1 2 3	
4	
2 1 3 5	

Подсказка по решению

Разобрана на лекции.

Задача 6D. 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

Подсказка по решению

Квадратная динамика. $f[i, j]$.

Задача 6Е. 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. Почти как НОП.

Задача 6F. Рюкзак [0.1 сек, 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.

Для искателей острых ощущений

Задача 6G. 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 6Н. 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 перебором «идём слева направо и...»