

## Содержание

<b>Must have</b>	<b>2</b>
Задача 6А. Зайчик [1, 256]	2
Задача 6В. Разложение на кубы [1, 256]	3
Задача 6С. Рюкзак [1, 256]	4
Задача 6D. Калькулятор [1, 256]	5
<b>Задачи здорового человека</b>	<b>6</b>
Задача 6Е. Три последовательности [1, 256]	6
Задача 6F. НПП [1, 256]	7
Задача 6G. Почтовые отделения [1, 256]	8
Для искателей острых ощущений	9
Задача 6H. Восстановление [1, 256]	9
Задача 6I. Наибольшая общая возрастающая [1, 256]	10

---

У вас не получается читать/выводить данные?

Воспользуйтесь примерами (c++) (python).

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

## Must have

### Задача 6А. Зайчик [1, 256]

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

stdin	stdout
4 ."".	2
5 .w"..	0
9 .www.www.	-1

### Задача 6В. Разложение на кубы [1, 256]

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

Важно посчитать динамику 1 раз, а не 10 000!

### Задача 6С. Рюкзак [1, 256]

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

#### Подсказка по решению

Самая простая версия рюкзака.

Лучше пишите с  $\mathcal{O}(S)$  памяти. И быстрее, и код короче.

### Задача 6D. Калькулятор [1, 256]

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

#### Подсказка по решению

В этой задаче самое важное – потренироваться восстанавливать ответ.

## Задачи здорового человека

### Задача 6Е. Три последовательности [1, 256]

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

#### Подсказка по решению

Не бойтесь задачи. Она такая же, как уже изученная вами.  
Только динамика теперь трёхмерная.

### Задача 6F. НПП [1, 256]

Для заданной числовой последовательности  $a_1, a_2, \dots, a_n$  требуется найти длину максимальной последовательнократной подпоследовательности.

Для последовательнократной подпоследовательности  $a_{k_1}, a_{k_2}, \dots, a_{k_t}$  ( $k_1 < k_2 < \dots < k_t$ ) верно, что  $a_{k_i} | a_{k_j}$  при  $1 \leq i < j \leq t$  (утверждение « $a|b$ » эквивалентно « $b$  кратно  $a$ »). Подпоследовательность из одного элемента полагается последовательнократной по определению.

#### Формат входных данных

В первой строке входного файла записано одно натуральное число  $N$  ( $1 \leq N \leq 1000$ ) — количество чисел в исходной последовательности. Далее следует  $N$  натуральных чисел, не превосходящих  $2 \cdot 10^9$  — сама последовательность.

#### Формат выходных данных

Вывести единственное число, равное искомому количеству.

#### Примеры

stdin	stdout
4 3 6 5 12	3

### Задача 6G. Почтовые отделения [1, 256]

Вдоль прямой дороги расположены деревни. Дорога представляется целочисленной осью, а расположение каждой деревни задается одним целым числом — координатой на этой оси. Никакие две деревни не имеют одинаковых координат. Расстояние между двумя деревнями вычисляется как модуль разности их координат.

В некоторых, не обязательно во всех, деревнях будут построены почтовые отделения. Деревня и расположенное в ней почтовое отделение имеют одинаковые координаты. Почтовые отделения необходимо расположить в деревнях таким образом, чтобы общая сумма расстояний от каждой деревни до ближайшего к ней почтового отделения была минимальной.

#### Формат входных данных

В первой строке содержатся два целых числа: количество деревень  $n$  ( $1 \leq n \leq 300$ ) и количество почтовых отделений  $m$  ( $1 \leq m \leq 30$ ),  $m \leq n$ . Вторая строка содержит  $n$  целых чисел в возрастающем порядке, являющихся координатами деревень. Для каждой координаты  $x$  верно  $1 \leq x \leq 10^4$ .

#### Формат выходных данных

Первая строка выходного файла должна содержать одно целое число — общую сумму расстояний от каждой деревни до её ближайшего почтового отделения. Вторая строка должна содержать  $m$  целых чисел в возрастающем порядке. Эти числа являются искомыми координатами почтовых отделений. Если для заданного расположения деревень есть несколько решений, необходимо найти любое из них.

#### Пример

stdin	stdout
10 5	9
1 2 3 6 7 9 11 22 44 50	2 7 22 44 50

#### Подсказка по решению

Разобрана на практике.  $\mathcal{O}(n^2m)$  точно получит ОК.



## Для искателей острых ощущений

### Задача 6Н. Восстановление [1, 256]

Денис обнаружил ошибку в своей программе, которая должна удалять все символы из строки кроме "(" и ")". Оказывается, некоторые скобки заменяются на что-то нечитаемое.

Теперь его заинтересовал вопрос, сколько различных правильных скобочных последовательностей могут являться результатом правильного алгоритма.

#### Формат входных данных

Единственная строка входного файла содержит строку из круглых скобок и знаков вопроса, где вопросами обозначены утраченные символы. Вопрос можно заменить на ровно одну любую скобку. Длина строки не превосходит 10 000, но может быть нечетной.

#### Формат выходных данных

Выведите одно число — количество различных скобочных последовательностей, удовлетворяющих шаблону Дениса, по модулю  $10^9 + 7$ .

#### Пример

stdin	stdout
(??()?)	2

#### Подсказка по решению

Вряд ли на питоне можно получить ОК. Сорри, так бывает.

**Задача 61. Наибольшая общая возрастающая [1, 256]**

Даны две последовательности чисел —  $a$  и  $b$ . Нужно найти наибольшую общую возрастающую подпоследовательность. Более формально: такие  $1 \leq i_1 < i_2 < \dots < i_k \leq a.n$  и  $1 \leq j_1 < j_2 < \dots < j_k \leq b.n$ , что  $\forall t : a_{i_t} = b_{j_t}, a_{i_t} < a_{i_{t+1}}$  и  $k$  максимально.

**Формат входных данных**

На первой строке целые числа  $n$  и  $m$  от 1 до 3000 — длины последовательностей. Вторая строка содержит  $n$  целых чисел, задающих первую последовательность. Третья строка содержит  $m$  целых чисел, задающих вторую последовательность. Все элементы последовательностей — целые неотрицательные числа, не превосходящие  $10^9$ .

**Формат выходных данных**

Выведите одно целое число — длину наибольшей общей возрастающей подпоследовательности.

**Пример**

stdin	stdout
6 5 1 2 1 2 1 3 2 1 3 2 1	2

**Подсказка по решению**

В этой задаче есть решение за  $\mathcal{O}(n^2)$  времени и  $\mathcal{O}(n)$  памяти.