

## Содержание

<b>Must have</b>	2
Задача 6А. Поколение комбинаторов [0.1, 256]	2
Задача 6В. Ацтекское золото [0.6, 256]	3
Задача 6С. Золотой песок [0.1, 256]	4
<b>Задачи здорового человека</b>	5
Задача 6D. Разбиения на слагаемые [0.1, 256]	5
Задача 6Е. Точки и отрезки [0.2, 256]	6
Задача 6F. Покрытие отрезков [0.2, 256]	7
Для искателей острых ощущений	8
Задача 6G. Анти-Фибоначчи [0.1, 256]	8
Задача 6H. Том Сойер и его друзья [0.2, 256]	9

---

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

Обратите внимание на GNU C++ компиляторы с суффиксом `inc`.

Подни можно пользоваться **дополнительной библиотекой** (`optimization.h`).

То есть, использовать быстрый ввод-вывод: **пример про числа и строки**.

И быструю аллокацию памяти (ускоряет `vector-set-map-весь-STL`): **пример**.

Для тех, кто хочет разобраться, как всё это работает.

Короткая версия быстрого ввода-вывода (**тык**) и короткая версия аллокатора (**тык**).

## Must have

### Задача 6А. Поколение комбинаторов [0.1, 256]

*Сочетанием* из  $n$  элементов по  $k$  называется убывающая последовательность из  $k$  чисел из диапазона от 1 до  $n$ .

Сгенерируйте все сочетания из  $n$  элементов по  $k$  в антилексикографическом порядке, т.е. так, что для любых двух выведенных сочетаний первые  $l$  чисел равны, а  $l+1$ -е в предыдущем больше, чем в следующем.

#### Формат входных данных

Во входном файле содержатся два целых числа  $n$  и  $k$ .  $1 \leq k \leq n \leq 15$ .

#### Формат выходных данных

В выходной файл выведите все сочетания из  $n$  элементов по  $k$  в антилексикографическом порядке, по одному сочетанию на строку.

#### Пример

stdin	stdout
3 2	3 2 3 1 2 1

### Задача 6В. Ацтекское золото [0.6, 256]

Вы грабите хранилище. У вас есть мешок, который выдержит суммарную массу  $W$ .  
В хранилище есть  $n$  ценных предметов, у каждого известен вес  $w_i$  и стоимость  $cost_i$ .  
Какую максимальную суммарную стоимость предметов можно унести с собой в мешке?

#### Формат входных данных

На первой строке  $n$  ( $3 \leq n \leq 25$ ) и  $W$  – вес рюкзака.  
На следующих  $n$  строках пары чисел  $w_i$   $cost_i$ , описывающие предметы.  
Числа  $W$ ,  $w_i$ ,  $cost_i$  – вещественные положительные.  $w_i, cost_i \leq 10$ .

#### Формат выходных данных

Выведите одно вещественное число с не менее чем 9 знаками после запятой – максимальную суммарную стоимость предметов можно унести с собой в мешке

#### Пример

stdin	stdout
3 10.0 4.5 5.0 8.0 9.0 4.5 4.1	9.100000000
5 20.00 7.08 8.18 1.27 4.46 2.28 4.53 8.02 9.69 8.77 4.68	26.860000000

#### Подсказка по решению

Это задача на рекурсивный перебор. Предполагается решение за  $2^n$ .

### Задача 6С. Золотой песок [0.1, 256]

Во время ограбления магазина вор обнаружил  $N$  ящичков с золотым песком. В ящичек под номером  $i$  песок имеет стоимость  $v_i$  и вес  $w_i$ . Чтобы унести награбленное, вор использует рюкзак. Требуется определить наибольшую суммарную стоимость песка, который может унести грабитель, если грузоподъемность рюкзака ограничена величиной  $W$ .

Из ящичков можно пересыпать любое количество песка, тогда отношение стоимости отсыпанного песка к стоимости всего ящичка будет равна отношению объема пересыпанного песка к объему всего ящичка.

#### Формат входных данных

В первой строке входного файла записаны два числа —  $N$  и  $W$  ( $1 \leq N \leq 1000$ ,  $0 \leq W \leq 2 \cdot 10^6$ ). Далее следует  $N$  строк по два целых числа в каждой. В  $i$ -ой строке записана стоимость  $v_i$  и вес  $w_i$  песка в  $i$ -ом ящичке. Все числа неотрицательные и не превосходят  $2 \cdot 10^6$ .

#### Формат выходных данных

Выведите искомую максимальную стоимость с точностью до 3 знаков после запятой.

#### Пример

stdin	stdout
3 50 60 20 100 50 120 30	180.000

## Задачи здорового человека

### Задача 6D. Разбиения на слагаемые [0.1, 256]

Перечислите все разбиения целого положительного числа  $N$  ( $1 \leq N \leq 40$ ) на целые положительные слагаемые. Разбиения должны обладать следующими свойствами:

1. Слагаемые в разбиениях идут в невозрастающем порядке.
2. Разбиения перечисляются в лексикографическом порядке.

#### Пример

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

### Задача 6Е. Точки и отрезки [0.2, 256]

Дано  $n$  отрезков на числовой прямой и  $m$  точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка  $x$  считается принадлежащей отрезку с концами  $a$  и  $b$ , если выполняется двойное неравенство  $\min(a, b) \leq x \leq \max(a, b)$ .

#### Формат входных данных

Первая строка содержит два целых числа  $n$  ( $1 \leq n \leq 10^5$ ) — число отрезков и  $m$  ( $1 \leq m \leq 10^5$ ) — число точек. В следующих  $n$  строках записаны по два целых числа  $a_i$  и  $b_i$  — координаты концов соответствующего отрезка. В последней строке записаны  $m$  целых чисел — координаты точек. Все числа во входном файле не превосходят по модулю  $10^9$ .

#### Формат выходных данных

В выходной файл выведите  $m$  чисел — для каждой точки выведите количество отрезков, в которых она содержится.

#### Примеры

stdin	stdout
2 2 0 5 7 10 1 6	1 0
1 3 -10 10 -100 100 0	0 0 1

### Задача 6F. Покрытие отрезков [0.2, 256]

Дано  $n$  отрезков на прямой. Выбрать минимальное количество точек на прямой так, чтобы каждому из данных отрезков принадлежала хотя бы одна точка. Или, иначе говоря, “покрыть отрезки отрезки минимальным числом точек”.

#### Формат входных данных

На первой строке  $n$  ( $1 \leq n \leq 100\,000$ ). Следующие  $n$  строк содержат пары целых чисел  $l_i, r_i$  ( $0 \leq l_i \leq r_i \leq 10^9$ ), задающие отрезки.

#### Формат выходных данных

На первой строке выведите  $k$  – количество точек, на следующей строке координаты точек в произвольном порядке. Если оптимальных ответов несколько, выведите любой.

#### Примеры

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

## Для искателей острых ощущений

### Задача 6G. Анти-Фибоначчи [0.1, 256]

Лёше надоели числа Фибоначчи. Всю последнюю неделю, когда он приходил на урок математики или информатики, учителя рассказывали что-то про числа Фибоначчи и задавали на дом задачки про них.

На этой неделе домашнее задание у Лёши — написать программу, которая по заданному целому положительному числу  $N$  находит количество способов разбить  $N$  на положительные целые слагаемые. Способы, отличающиеся лишь порядком слагаемых, считаются одинаковыми. К примеру, для  $N = 4$  это количество способов — 5:

$$\begin{aligned} N &= 4 \\ &= 3 + 1 \\ &= 2 + 2 \\ &= 2 + 1 + 1 \\ &= 1 + 1 + 1 + 1 \end{aligned}$$

Поскольку Лёше не нравятся числа Фибоначчи, он решил написать программу, которая считает только такие разбиения, в которых среди слагаемых нет чисел Фибоначчи. Более того, разбиения, в которых количество слагаемых является числом Фибоначчи, Лёша тоже решил не считать.

Помогите Лёше написать такую программу.

#### Формат входных данных

В первой строке входного файла записано целое число  $N$  ( $1 \leq N \leq 50$ ).

#### Формат выходных данных

Выведите в выходной файл одно число — количество таких разбиений  $N$  на положительные целые слагаемые, что ни какое-либо из слагаемых, ни их количество не являются числами Фибоначчи.

#### Пример

stdin	stdout
4	0



### Задача 6Н. Том Сойер и его друзья [0.2, 256]

Друзья Тома Сойера по очереди красят забор разными красками. Каждый из них красит несколько идущих подряд секций забора в определенный цвет, при этом используемые цвета могут повторяться. Новая краска ложится поверх старой. Для каждой краски вычислите количество секций, которые будут покрашены этой краской после того, как все друзья закончат работу.

#### Формат входных данных

В первой строке входного файла содержатся два целых числа:  $N$  ( $1 \leq N \leq 10^9$ ) и  $K$  ( $1 \leq K \leq 50\,000$ ) — количество секций в заборе и количество различных красок соответственно.

Во второй строке содержится единственное число  $M$  ( $0 \leq M \leq 50\,000$ ) — количество друзей Тома Сойера.

Далее следуют  $M$  строк: в  $i$ -ой строке содержится информация о работе друга, который красил забор  $i$ -ым по счету, а именно 3 целых числа  $c_i, l_i, r_i$  ( $1 \leq c_i \leq K, 1 \leq l_i \leq r_i \leq N$ ) — номер краски, которую использовал  $i$ -й друг, номер первой и номер последней покрашенной секции соответственно.

#### Формат выходных данных

Выведите в единственную строку выходного файла  $K$  целых чисел:  $i$ -ое число должно быть равно количеству секций, покрашенных  $i$ -й краской.

#### Пример

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