

## Задача А. Двоичное дерево

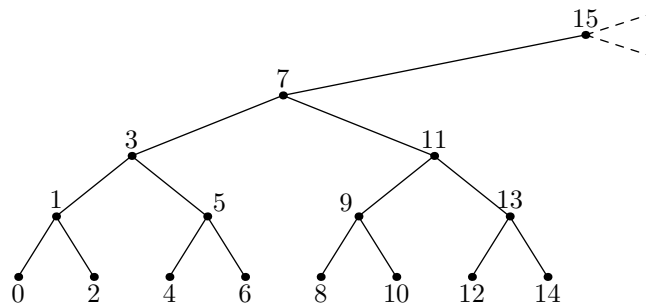
Имя входного файла: `binary.in`  
Имя выходного файла: `binary.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Never underestimate the power of two!

цитата **7ania7** на [www.topcoder.com](http://www.topcoder.com)

На Кафедре Оптимизации Деревьев Отрезков и (двоичных) Куч (сокр. КОДОК) знают, что номера вершинам в двоичных деревьях удобно присваивать сверху вниз: корень имеет номер 1, его дети — 2 и 3, и так далее. Действительно, удобно — чтобы найти «отца» какой-либо вершины, нужно всего лишь её номер поделить пополам.

Но в математике, как обычно, всё задом наперёд, и в этой задаче вершины нумеруют слева направо (см. рисунок). Ваша же задача проста — найти сумму номеров вершин на пути от вершины  $a$  до вершины  $b$ . Считайте, что корень дерева имеет номер 1 427 247 692 705 959 881 058 285 969 449 495 136 382 746 623



### Формат входного файла

В первой и единственной строке ввода даны два числа:  $a$  и  $b$  — это номера вершин ( $0 \leq a, b \leq 10^{15}$ ).

### Формат выходного файла

В единственной строке вывода должно быть одно число — сумма номеров вершин на пути от вершины с номером  $a$  до вершины с номером  $b$ .

### Примеры

<code>binary.in</code>	<code>binary.out</code>
1 5	9
3 4	12

## Задача В. Номер скобочной последовательности

Имя входного файла: `brackets2num.in`  
Имя выходного файла: `brackets2num.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Во входном файле задана правильная скобочная последовательность. Выведите в выходной ее номер в лексикографическом порядке среди всех правильных скобочных последовательностей с таким же количеством открывающихся скобок, «(» < «)». Последовательности занумерованы, начиная с 0. Количество открывающихся скобок в последовательности — от 1 до 20.

### Пример

<code>brackets2num.in</code>	<code>brackets2num.out</code>
<code>((()))()</code>	3

## Задача С. ПСП с двумя типами скобок по номеру

Имя входного файла: `num2brackets2.in`  
Имя выходного файла: `num2brackets2.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Во входном файле заданы числа  $n$  и  $k$ . Выведите в выходной файл  $k$ -ю в лексикографическом порядке правильную скобочную последовательность среди всех правильных скобочных последовательностей с двумя типами скобок с  $n$  открывающимися скобками, упорядоченных в лексикографическом порядке, «(» < «)» < «[» < «]». Последовательности занумерованы, начиная с 0.  $1 \leq n \leq 20$ . Искомая последовательность существует.

### Пример

<code>num2brackets2.in</code>	<code>num2brackets2.out</code>
4 100	( [ ] ) ( ) [ ]

## Задача D. Сочетание по номеру

Имя входного файла: `num2choose.in`  
Имя выходного файла: `num2choose.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Во входном файле заданы числа  $n$ ,  $k$  и  $m$ . Выведите в выходной файл  $m$ -е в лексикографическом порядке сочетание по  $k$  из чисел от 1 до  $n$ . Сочетания занумерованы, начиная с 0.  $1 \leq k \leq n \leq 30$ ,  $0 \leq m \leq \binom{n}{k} - 1$ .

### Пример

<code>num2choose.in</code>	<code>num2choose.out</code>
4 2 3	2 3

## Задача Е. Номера страниц

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

Петя Васечкин решил пронумеровать страницы в своей тетради числами от 1 до  $N$ . Определите количество нулей, единиц, ..., девяток, которые ему потребуются.

### Формат входного файла

Целое число  $N$  ( $1 \leq N \leq 10^9$ ).

### Формат выходного файла

Выведите 10 строк, первая из которых содержит необходимое количество нулей, вторая – единиц, ..., десятая – девяток.

### Примеры

<code>pages.in</code>	<code>pages.out</code>
12	1 5 2 1 1 1 1 1 1 1

## Задача F. Номер разбиения на слагаемые

Имя входного файла: `part2num.in`  
Имя выходного файла: `part2num.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Рассмотрим все разбиения числа  $n$  на слагаемые, в каждом разбиении упорядочим их в порядке не убывания. Будем считать, что разбиение  $a_1 + a_2 + \dots + a_n$  лексикографически меньше  $b_1 + b_2 + \dots + b_m$ , если для некоторого  $k \forall j \leq k : a_j = b_j$  и либо  $k = n$ , либо  $a_{k+1} < b_{k+1}$ . Во входном файле задано разбиение на слагаемые. Выведите номер этого разбиения, среди всех разбиений упорядоченных лексикографически. Разбиения нумеруются с 0. Гарантируется, что в разбиении слагаемые упорядочены в порядке не убывания, и  $1 \leq n \leq 100$ .

### Пример

<code>part2num.in</code>	<code>part2num.out</code>
2+2	3

## Задача G. Фотография

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

Антон расставляет детей для общей фотографии. Чтобы выстроить их наилучшим образом, он решил, не мудрствуя лукаво, попробовать все способы. Для этого он выписал их на листик в алфавитном порядке. И стал пробовать. Но на каком-то по счёту расположении детей список потерялся, и теперь требуется его вывести заново как можно быстрее. Понятно, что в целях экономии времени, восстановить его нужно только с текущего положения (а точнее, следующего за ним).

Сделайте это, пожалуйста!

### Формат входного файла

В первой строке входного файла записаны имена детей в текущей расстановке.

Как известно, имена детей в России не очень длинные и состоят из букв. Для Вашего удобства, все они записаны буквами латинского алфавита (с заглавной, естественно).

### Формат выходного файла

В выходной файл выведите требуемый список, по одному расположению детей на строке.

Вы скажете, что это невозможно? Но нет, вам повезло, и сегодня много отсутствующих, так что все дети в списке имеют различные имена.

### Примеры

<code>photo.in</code>	<code>photo.out</code>
Vika Polina Alina Sasha	Vika Polina Sasha Alina Vika Sasha Alina Polina Vika Sasha Polina Alina

## Задача Н. Фотография — 2

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

Антон понял, что его группа может достигать 20-ти человек (но не более), и список что-то великоват. Поэтому, на этот раз, он просит Вас вывести около 10 строк из итогового списка.

Чтобы строчки располагались равномерно по оставшейся части списка, предлагается сделать следующее. Допустим, номер текущего положения  $s$  ( $1 \leq s \leq n$ ), а всего в списке было  $n$  строк. Тогда осталось  $n - s$  строк. Вычислим величину  $\Delta = \lceil \frac{n-s}{10} \rceil$  (при делении округляем вверх). А после этого выведем строчки  $s + \Delta, s + 2\Delta, s + 3\Delta, \dots$  и так, пока список не кончится.

Сделайте это, пожалуйста!

### Формат входного файла

В первой строке входного файла записаны имена детей в текущей расстановке.

Как известно, имена детей в России не очень длинные и состоят из букв. Для Вашего удобства, все они записаны буквами латинского алфавита (с заглавной, естественно).

### Формат выходного файла

В выходной файл выведите требуемые строчки из списка, по одной строчке в каждой строке выходного файла. Все дети, по-прежнему, имеют различные имена.

### Примеры

<code>photo-2.in</code>	<code>photo-2.out</code>
Vika Polina Alina Sasha	Vika Polina Sasha Alina Vika Sasha Alina Polina Vika Sasha Polina Alina



## Задача I. Слова

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

Антон очень любит День Анаграмм в ЛКШ и в каждой смене его проводит. Целую ночь все преподаватели заняты увеселительной процедурой изменения детских бейджиков и объявлений. В помощь недосыпающим даже была написана программа, но и она несильно помогает. В этой задаче вам нужно посчитать насколько всё плохо.

Кстати, вчера на сайте [lksh.ru](http://lksh.ru) была опубликована вступительная работа, если вы ещё не в 11 классе, не забудьте поступить и приехать к нам летом.

Ой, и как обычно, эта часть условия не несла смысловой нагрузки, а всё что нужно для решения задачи написано ниже.

### Формат входного файла

В единственной строке входного файла записано одно слово из маленьких латинских букв. Его длина не превосходит 30.

### Формат выходного файла

В выходной файл выведите количество различных слов, которые можно составить из букв данного слова.

### Примеры

<code>words.in</code>	<code>words.out</code>
alina	60
polina	720
sasha	30
vika	24

## Задача J. Слова — 2 (New!)

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

Антон обратил внимание, что количество слов, которые можно составить из букв данного слова очень большое. Так что лучше выведите остаток от деления данного количества на данное Вам число  $M$ .

Чуть позже ему пришлось обратить внимание на то, что вычислить  $1000!$  для Питона — сущий пустяк, а зная другие языки программирования, он может вычислить ответ даже для строчек длины  $10^6$ . А вы сможете?

### Формат входного файла

В первой строке входного файла записано число  $1 \leq M \leq 10^9$ . А во второй строке одно слово из маленьких латинских букв. Его длина не превосходит  $10^6$ .

### Формат выходного файла

В выходной файл выведите количество различных слов, которые можно составить из букв данного слова, взятое по модулю  $M$ .

### Примеры

words-2.in	words-2.out
30 alina	0
30 polina	0
165 sasha	30
30 vika	24