

Задача 1. Сравнение элементов

Источник: базовая
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дан массив из N чисел: b_1, b_2, \dots, b_N .

Требуется для каждого элемента массива b_i посчитать количество элементов b_j , стоящих правее ($i < j$), но меньших него ($b_i > b_j$).

Формат входных данных

В первой строке записано целое число N ($1 \leq N \leq 100$).

Во второй строке через пробел записано N целых положительных чисел — значения элементов массива. Гарантируется, что значения элементов не превосходят 100.

Формат выходных данных

Выведите N целых чисел через пробел: по порядку для каждого элемента входного массива выведите количество элементов, удовлетворяющих описанному выше условию.

Примеры

input.txt	output.txt
5 4 3 5 1 2	3 2 2 0 0
3 3 2 1	2 1 0
4 1 2 3 3	0 0 0 0

Пояснение к примеру

В первом примере:

- Правее числа 4 стоят числа 3, 5, 1 и 2, **три** из которых имеют значение меньше 4.
- Правее числа 3 стоят числа 5, 1 и 2, **два** из которых имеют значение меньше 3.
- Правее числа 5 стоят числа 1 и 2, **два** из которых имеют значение меньше 5.
- Для чисел 1 и 2 справа нет чисел, имеющих значение меньше.

Задача 2. Суммы k -ых

Источник: базовая
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дан массив из N чисел a_1, a_2, \dots, a_N .

Требуется вывести N чисел t_k ($k = 1, \dots, N$), где t_k — сумма элементов массива с шагом k :

$$t_k = \sum_{j=1}^{N/k} a_{j \cdot k}$$

Т.е. t_1 равняется сумме всех элементов массива, t_2 равняется сумме каждого второго элемента массива, t_3 равняется сумме каждого третьего элемента и т.д.

Формат входных данных

В первой строке записано целое число N ($1 \leq N \leq 10^5$).

Во второй строке через пробел записано N целых чисел a_i ($1 \leq a_i \leq 10^4$).

Формат выходных данных

Выведите N строк: в k -й строке выведите значение t_k .

Пример

input.txt	output.txt
6	24
4 3 5 1 2 9	13
	14
	1
	2
	9

Пояснение к примеру

$$t_1 = a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 4 + 3 + 6 + 1 + 2 + 9 = 24$$

$$t_2 = a_2 + a_4 + a_6 = 3 + 1 + 9 = 13$$

$$t_3 = a_3 + a_6 = 5 + 9 = 14$$

$$t_4 = a_4 = 1$$

$$t_5 = a_5 = 2$$

$$t_6 = a_6 = 9$$

Задача 3. Гистограмма

Источник: базовая
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дан массив из N чисел a_0, a_1, \dots, a_{N-1} .

Постройте гистограмму значений элементов массива: для каждого значения подсчитайте сколько раз оно встречается в массиве.

Формат входных данных

В первой строке записано целое число N ($1 \leq N \leq 10^5$).

Во второй строке через пробел записано N целых чисел a_i ($1 \leq a_i \leq 10^4$).

Формат выходных данных

Для каждого значения, которое встречается в массиве хотя бы раз, выведите сколько раз оно присутствует среди элементов массива в формате: “значение: количество”.

Значения требуется выводить в порядке возрастания.

Для вывода двух целых чисел, разделённых двоеточием с пробелом, удобно использовать функцию `printf` со следующей форматной строкой:

```
printf("%d: %d", value, count);
```

Пример

input.txt	output.txt
10 3 4 5 10 3 4 10 1 1 3	1: 2 3: 3 4: 2 5: 1 10: 2

Задача 4. Сумма

Источник: основная
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дан массив из N чисел a_0, a_1, \dots, a_{N-1} .

Требуется найти два индекса L и R ($L \leq R$) таких, что сумма $a_L + a_{L+1} + \dots + a_{R-1} + a_R$ будет максимальной.

Формат входных данных

В первой строке записано целое число N ($1 \leq N \leq 5\,000$).

Во второй строке через пробел записано N целых чисел, каждое из которых по модулю не превосходит 10 000.

Формат выходных данных

Выведите три целых числа через пробел: L , R и сумму элементов с индексами от L до R .

Если существует несколько вариантов выбрать такие L и R , что сумма элементов будет максимальной, то требуется вывести вариант с минимальным L . Если существует несколько с минимальным L , то среди таких требуется выбрать вариант с минимальным R .

Примеры

input.txt	output.txt
3 1 2 3	0 2 6
5 1 -2 3 4 5	2 4 12
4 2 -2 1 1	0 0 2

Задача 5. Функция

Источник:	основная
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

Пусть P — множество целых чисел от 1 до N . Задано отношение R на множестве $P \times P$. Отношение задаётся списком принадлежащих ему элементов: множеством пар $(x, y) \in P \times P$.

Для заданного отношения R требуется определить:

1. Является ли R функцией: $\forall x \in P: (x, u) \in R$ и $(x, v) \in R \Rightarrow u = v$ — отсутствует многозначность.
2. Является ли R всюду определённой функцией: R является функцией и $\forall x \in N \exists y \mid (x, y) \in R$ — значение определено на всём множестве P .
3. Является ли R инъекцией: $(x, u) \in R$ и $(y, u) \in R \Rightarrow x = y$.
4. Является ли R сюръекцией: $\forall u \in N \exists x \mid (x, u) \in R$.
5. Является ли R биекцией: отношение R и инъективно, и сюръективно.

Формат входных данных

Первая строка входного файла содержит два целых числа N и M , записанных через пробел — размер множества ($1 \leq N \leq 300$) и количество пар ($1 \leq M \leq N^2$).

В следующих M строках записано по два целых числа x и y ($1 \leq x, y \leq N$) — элементы отношения R . Гарантируется, что все пары различны.

Формат выходных данных

Если отношение R не удовлетворяет ни одному из описанных свойств, то выведите число 0.

В противном случае выведите через пробел в порядке увеличения номера свойств, которыми обладает отношение R .

Примеры

input.txt	output.txt
3 5 1 2 3 3 1 3 1 1 2 1	0
5 2 1 3 2 3	1
5 2 1 3 2 5	1 3
3 3 1 1 2 3 3 2	1 2 3 4 5

Задача 6. Разница множеств

Источник: основная
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дано два массива целых чисел A и B .

Требуется найти все такие значения элементов массива A , которых нет среди элементов массива B .

Формат входных данных

В первой строке записано целое число N ($1 \leq N \leq 10^5$) — количество элементов массива A .

Во второй строке через пробел записано N неотрицательных целых чисел, каждое из которых не превосходит 10^5 — элементы массива A .

В следующих двух строках в аналогичном формате записаны элементы массива B .

Формат выходных данных

В первой строке выведите одно целое число — количество значений, удовлетворяющих описанному условию.

Во второй строке выведите все такие значения в порядке возрастания.

Примеры

input.txt	output.txt
7 1 2 3 3 6 8 8	3 2 6 8
4 1 3 7 9	
3 1 2 3 3 3 2 1	0

Задача 7. Навести порядок

Источник: повышеннoй сложности*
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дан массив, состоящий из N целых чисел.

Требуется вывести эти N чисел в выходной файл, но в строго определённом порядке. Каждое следующее выведенное число должно быть больше предыдущего или равно ему.

Формат входных данных

В первой строке записано целое число N ($1 \leq N \leq 10^4$) — количество элементов массива.

Во второй строке через пробел записано N целых чисел, каждое из которых по модулю не превосходит 10^5 .

Формат выходных данных

В единственной строке выведите все N элементов массива через пробел в указанном порядке.

Пример

input.txt	output.txt
5 1 4 3 5 3	1 3 3 4 5

Задача 8. Системы счисления

Источник: повышеннoй сложности
Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: разумное

Дано число N в p -ичной системе счисления.

Требуется выполнить перевод числа N в q -ичную систему счисления.

Формат входных данных

В первой строке через пробел записаны три числа p , q и N ($2 \leq p, q \leq 36$). Гарантируется, что значение числа N в десятичной системе счисления не превосходит 10^9 .

Для записи цифр, значения которых в десятичной системе счисления имеют значения от 10 до 36, используются строчные латинские буквы 'a', 'b', ..., 'z'.

Формат выходных данных

Выведите число N в q -ичной системе счисления.

Примеры

input.txt	output.txt
2 16 101010	2a
7 20 22	g
20 7 g	22

Задача 9. Буквы алфавита

Источник:	повышенной сложности
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

Вам дана строка, состоящая из строчных букв латинского алфавита. Все буквы в ней различны.

Требуется переставить буквы данной строки так, чтобы получившаяся строка была лексикографически больше исходной.

Поскольку в данной постановке задача слишком простая и может иметь не единственное решение, то вам требуется среди всех таких строк выбрать лексикографически минимальную.

Строка s , состоящая из символов s_0, s_1, \dots, s_n , считается *лексикографически меньше* строки t , состоящей из символов t_0, t_1, \dots, t_n , если существует индекс k такой, что:

- $s_i = t_i$ для всех $i = 0, 1, \dots, k - 1$;
- $s_k < t_k$.

Иными словами, лексикографическое сравнение строк - это привычное нам сравнение слов “по алфавиту”, когда мы находим первую букву, в которой две строки различаются, и на основании этой буквы делаем вывод о том, какое из слов “меньше”. Лексикографическое сравнение окружает нас повсюду: его можно найти в порядке людей в списках групп, в порядке номеров в телефонной книге, и т.д.

Формат входных данных

В первой строке записано целое число N ($2 \leq N \leq 26$) — количество символов в строке.

Во второй строке через пробел записано N строчных букв латинского алфавита. Гарантируется, что все буквы различны.

Формат выходных данных

Выведите через пробел символы требуемой строки.

Гарантируется, что требуемая перестановка существует.

Примеры

input.txt	output.txt
5 a b c d e	a b c e d
3 q z w	w q z

Задача 10. Шоколадная палочка Фрикс

Источник:	космической сложности
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Мухи любят уединение. А ещё они очень любят шоколад, в особенности шоколадные палочки от известной фирмы “Фрикс”. Когда новая муха хочет сесть на палочку, она старается выбрать себе место подальше от остальных мух, сидящих на ней.

Представим шоколадную палочку в виде отрезка $[0, L]$ на координатной прямой. Будем считать, что каждая сидящая на палочке муха занимает некоторый интервал длины W . В любой момент времени все интервалы, занимаемые мухами, не пересекаются и не выходят за пределы палочки.

Допустим, на палочке уже сидит некоторое количество мух, и хочет сесть ещё одна. Если на палочке нет подходящего свободного места, то муха улетает. В противном случае она садится таким образом, чтобы расстояние от неё до других мух и концов палочки было наибольшим.

Формально говоря, место, на которое садится новая муха, определяется следующим образом. Пусть Y — множество, равное объединению всех интервалов, на которых уже сидят мухи, и концевых точек палочки $\{0, L\}$. Интервал, который занимает вновь прилетевшая муха, таков, что расстояние от его центра до множества Y максимально. Если на палочке есть несколько таких точек, то в качестве центра из них выбирается точка с минимальной координатой.

Изначально нужно разместить на палочке ровно N мух. Далее к палочке по одному будут подлетать мухи и садиться на неё по вышеописанным правилам. Это будет продолжаться до тех пор, пока на палочке не закончится свободное место.

В задаче требуется найти два варианта изначальной рассадки мух. В первом варианте окончательное количество сидящих на палочке должно быть минимально возможным, в во втором — максимально возможным.

Формат входных данных

В первой строке входного файла задано два целых числа N и W ($1 \leq N \leq 100\,000$, $1 \leq W \leq 10^9$). Во второй строке записано целое число L ($1 \leq L \leq 10^{17}$).

Гарантируется, что можно разместить N мух на палочке.

Формат выходных данных

В первую строку выходного файла нужно вывести два целых числа — минимально возможное количество уместившихся на палочке и максимально возможное количество. Далее необходимо вывести два варианта рассадки — рассадку, при которой достигается минимальное количество, а затем рассадку, при которой количество мух максимально.

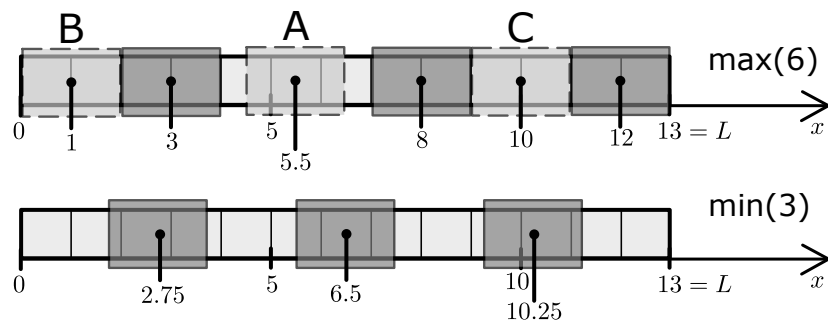
Каждая рассадка должна занимать N строк, по одному числу в каждой. Каждое число — это координата центра интервала, занимаемого мухой. Все числа в одной рассадке должны идти в порядке возрастания. Разрешается выводить вещественные числа, но не более чем с 9-ю десятичными знаками после запятой. Гарантируется, что существуют оптимальные рассадки, которые можно представить таким образом.

Пример

input.txt	output.txt
3 2 13	3 6 2.75 6.5 10.25 3 8.0 12.00

Комментарий

На картинке изображены обе рассадки из примера.



В минимальной рассадке на палочке есть четыре свободных отрезка, однако все они имеют длину 1.75, поэтому сесть больше никто не может.

В максимальной рассадке сначала сидят три мухи. Потом подлетают ещё три, их места обозначены буквами **A**(5.5), **B**(1), **C**(10) в порядке их появления.