

## Задача A. Dictionary

Имя входного файла: `dict.in`  
Имя выходного файла: `dict.out`  
Ограничение по времени: 4 seconds  
Ограничение по памяти: 64 Mebibytes

Байтланские учёные установили, что слова байтландского языка образуются путём конкатенации двух других (возможно, совпадающих) слов этого языка, при этом одно слово может иметь несколько вариантов происхождения. По заданному словарю языка требуется сказать, сколькими способами можно получить каждое из его слов.

### Формат входного файла

В первой строке входного файла находится целое число  $N$  ( $1 \leq N \leq 10^5$ ) — количество слов в словаре. В следующих  $N$  строках записаны сами слова, состоящие из строчных латинских букв, их длина не превышает 50 символов. Все слова в словаре различны. Можно использовать одно слово дважды.

### Формат выходного файла

Выведите  $N$  строк, содержащих по одному целому числу. Число в  $i$ -й строке должно означать количество способов получить  $i$ -ое слово путем соединения каких-либо двух других слов из этого же словаря.

### Пример

<code>dict.in</code>	<code>dict.out</code>
5	0
x	1
xx	2
xxx	3
xxxx	4
xxxxx	

## Задача В. New fabric of HAL

Имя входного файла: `fabric.in`  
Имя выходного файла: `fabric.out`  
Ограничение по времени: 2 seconds  
Ограничение по памяти: 64 Mebibytes

Компания HAL, производящая суперкомпьютеры, решила открыть филиал в Байтландии. Для этого компания приобрела в производственной зоне несколько участков, принадлежавших ранее различным компаниям. Естественно, что каждый участок был огорожен стенами. При этом:

- Участки имеют форму  $n$ -угольника, где  $3 \leq n \leq 8$ . Стороны  $n$ -угольника при этом являются стенами.
- Некоторые участки могут иметь общую стену.

Для каждой стены известно, сколько будет стоить её снос, при этом сносить можно только стену целиком. Руководство компании хочет, затратив минимальные средства, добиться того, чтобы с любой точки территории завода можно было попасть в любую другую. При этом путь может проходить и по «внешней» территории, которая не принадлежит компании.

Ваша задача — определить минимальную сумму, которую необходимо затратить на выполнение распоряжения руководства.

### Формат входного файла

В первой строке входного файла задано целое число  $M$  — количество купленных участков ( $1 \leq M \leq 100$ ). В последующих  $M$  строках перечислены сами участки. Описание  $i$ -го участка начинается с числа  $n_i$  — количества сторон у данного участка ( $3 \leq n_i \leq 8$ ). Далее следуют  $n_i$  целых чисел  $a_j$  ( $1 \leq a_j \leq 1000$ ) — номера вершин, составляющих этот участок. Вершины перечислены в порядке обхода периметра соответствующего участка. При этом нумерация вершин для всех участков сквозная, то есть если в описании двух участков встретились вершины с равными номерами, то данная вершина для этих участков общая, если встретилась пара соседних (включая первую и  $n_i$ -ю) вершин, то соответствующая стена для этих участков общая. Далее следуют  $n_i$  целых чисел  $c_j$  ( $1 \leq c_j \leq 5000$ ) — соответственно стоимость сноса стены  $(a_j, a_{j+1})$  (при  $j = n_i - (a_{n_i}, a_1)$ ).

Гарантируется, что «внешняя» территория является связной.

### Формат выходного файла

Выведите одно число — минимальную сумму, которую необходимо потратить на снос стен для того, чтобы с любой точки купленной компанией территории можно было попасть в любую другую.

### Пример

<code>fabric.in</code>	<code>fabric.out</code>
4 3 3 2 1 4 7 6 4 5 4 2 1 2 7 7 6 4 5 6 7 4 9 8 4 2 5 8 7 4 2 3 7 4 7 4 7	10
2 3 1 2 3 9 1 11 3 1 2 14 9 2 12	3

## Задача C. Internet access

Имя входного файла: `inet.in`  
Имя выходного файла: `inet.out`  
Ограничение по времени: 3 seconds  
Ограничение по памяти: 256 Mebibytes

К выставке WiFiExpo в Байтландии построили огромный выставочный комплекс, в котором  $M$  горизонтальных и  $N$  вертикальных рядов. Соседние ряды расположены на расстоянии 1.

На пересечении каждого вертикального и горизонтального ряда находится интернет-кафе. В  $K$  из этих кафе установлены точки доступа, для каждой из которых задана её пропускная способность  $B_i$  и радиус зоны видимости  $R_i$ .

При этом сеть устроена так, что во всех точках, расстояние от которых до  $i$ -го кафе с точкой доступа не превышает  $R_i$ ,  $i$ -е соединение идёт на полной скорости, а в остальных точках данное соединение не подключается.

Вася приехал на выставку WiFiExpo в командировку сразу от нескольких институтов, поэтому ему необходимо отправлять и принимать большие объёмы данных. Установленная на ноутбуке система позволяет Васе подключаться ко всем доступным в данном месте локальным сетям так, что получившаяся пропускная способность равна сумме пропускных способностей используемых сетей, так что Вася решил найти интернет-кафе, в котором пропускная способность будет максимальной.

Вычислите эту пропускную способность, а также количество интернет-кафе на выставке, в которых она достигается.

### Формат входного файла

В первой строке входного файла задано целое число  $M$  — количество горизонтальных рядов выставочного комплекса ( $1 \leq M \leq 3 \cdot 10^4$ ). Во второй строке входного файла задано целое число  $N$  — количество вертикальных рядов выставочного комплекса ( $1 \leq N \leq 1000$ ). В третьей строке входного файла задано целое число  $K$  — количество интернет-кафе, в которых установлены точки доступа ( $1 \leq K \leq 1000$ ). В последующих  $K$  строках описаны точки доступа. В  $i$ -й из этих строк содержится четыре целых числа:  $x_i$  и  $y_i$  — номер соответственно вертикального и горизонтального ряда, на пересечении которых находится интернет-кафе, где размещена  $i$ -я точка доступа ( $1 \leq x_i \leq N$ ,  $1 \leq y_i \leq M$ ),  $R_i$  — радиус зоны видимости  $i$ -й точки доступа ( $1 \leq R_i \leq 3 \cdot 10^4$ ) и  $B_i$  — пропускная способность  $i$ -й точки доступа ( $1 \leq B_i \leq 1000$ ).

### Формат выходного файла

В первой строке выходного файла выведите максимальную суммарную пропускную способность, которой может добиться Вася в каком-нибудь интернет-кафе. Во второй строке выходного файла выведите количество интернет-кафе, в которых это значение пропускной способности может быть достигнуто.

### Пример

<code>inet.in</code>	<code>inet.out</code>
6	10
10	9
3	
2 5 4 4	
6 1 4 6	
9 3 2 3	

## Задача D. Rating

Имя входного файла: `rating.in`  
Имя выходного файла: `rating.out`  
Ограничение по времени: 2 seconds  
Ограничение по памяти: 64 Mebibytes

Разработанная в Бейтландии новая платформа для смартфонов Paganoid включает в себя систему выбора приложений Restore. Одним из элементов Restore является рейтинг приложений. Рейтинг реализован работает следующим образом. Каждый пользователь выставляет рейтинг тем приложениям, которые он установил у себя на устройстве.

В новой версии Paganoid функции системы выбора приложений расширены — планируется разработка модуля, который рекомендовал бы пользователю новые приложения. Модуль должен для каждого пользователя выдавать все приложения, которые не были им установлены (и оценены) со средним рейтингом, попадающим в интервал между максимальной и минимальной оценкой, выставленной этим пользователем при оценке установленных на его смартфон приложений.

### Формат входного файла

В первой строке входного файла заданы два целых числа  $M$  и  $N$  — количество пользователей и количество приложений соответственно ( $1 \leq M, N \leq 10^4$ ). Далее следуют  $M$  строк.  $i$ -я строка задаёт список приложений, выбранных  $i$ -м пользователем. Сначала идёт целое число  $n_i$  — количество приложений, выбранных  $i$ -м пользователем ( $1 \leq n_i \leq N$ ). Затем следуют  $n_i$  пар целых чисел, задающих приложения.

Каждое приложение задаётся в формате  $(a_{ij}, r_{ij})$ , где  $a_{ij}$  — номер  $j$ -го приложения, выбранного  $i$ -м пользователем ( $0 \leq a_{ij} \leq N - 1$ ), а  $r_{ij}$  — рейтинг  $j$  приложения по мнению  $i$ -го пользователя ( $0 \leq r_{ij} \leq 1023$ ). Гарантируется, что каждое приложение оценено хотя бы одним пользователем и что каждый пользователь оценил хотя бы одно приложение.

### Формат выходного файла

Для каждого пользователя в порядке следования пользователей во входном файле выведите список рекомендованных приложений в формате: номер пользователя, двоеточие, и далее — список рекомендованных приложений, отсортированный по возрастанию номера приложения. Гарантируется, что объём выходного файла не превышает 1 MiB.

### Примеры

<code>rating.in</code>	<code>rating.out</code>
2 2 1 0 7 1 1 7	0: 1 1: 0
4 5 3 0 150 1 185 4 170 3 0 120 2 130 3 130 4 0 135 1 130 2 130 3 130 3 1 130 2 130 3 199	0: 3 1: 2: 3: 0 4

## Задача E. Road

Имя входного файла: `road.in`  
Имя выходного файла: `road.out`  
Ограничение по времени: 2 seconds  
Ограничение по памяти: 64 Mebibytes

Кольцевая дорога вокруг столицы Байтландии представляет собой окружность длиной  $10^6$ . На кольцевой дороге находятся  $N$  развязок. Для предотвращения возможных пробок было принято решение разместить на дороге  $K$  постов дорожной полиции так, чтобы максимальное расстояние от перекрёстка до ближайшего к нему поста было как можно меньше. Ваша задача — вычислить это расстояние.

### Формат входного файла

В первой строке входного файла задано целое число  $N$  — количество развязок ( $1 \leq N \leq 1000$ ). В последующих  $N$  строках содержится по одному целому числу  $a_i$  — расстояние по кольцевой линии от самой северной точки развязки до  $i$ -го перекрёстка, отсчитываемое по часовой стрелке ( $0 \leq a_i < 10^6$ ). При этом  $a_i = a_j$  тогда и только тогда, когда  $i = j$ . В  $N + 2$ -й строке задано число  $K$  ( $1 \leq K \leq 1000$ ) — количество постов дорожной полиции, которые нужно разместить.

### Формат выходного файла

Одно число — минимальное значение максимального расстояния от перекрёстка до ближайшего к нему поста дорожной полиции, выведенное с точностью 0.01.

### Примеры

road.in	road.out
4 3 96000 97000 106000 2	5000.0
4 0 250000 500000 750000 4	0

## Задача F. Triangle

Имя входного файла: `triangle.in`  
Имя выходного файла: `triangle.out`  
Ограничение по времени: 2 seconds  
Ограничение по памяти: 64 Mebibytes

На гербе одного из городов Байтландии изображён прямоугольный треугольник, в который вписана ломаная, строящаяся следующим образом: Первое звено является высотой, опущенной из вершины прямого угла на гипотенузу, затем из точки пересечения опускается перпендикуляр на больший катет, затем опять на гипотенузу и так до бесконечности. Требуется вычислить длину получившейся ломаной.

### Формат входного файла

В первой строке входного файла находятся два целых числа  $A$  и  $B$ , разделенных пробелами ( $1 \leq A, B \leq 1000$ ).

### Формат выходного файла

Выведите длину получившейся ломаной с точностью до  $10^{-6}$ .

### Пример

<code>triangle.in</code>	<code>triangle.out</code>
4 3	12.0