

## А. Производство

Ограничение по времени: **1000 миллисекунд**  
 Ограничение по памяти: **65000 кибибайт**

На заводе имеется  $N$  ( $1 \leq N \leq 20$ ) станков и для каждого станка имеется чётко прописанный технологический процесс. Известно, что  $i$ -й станок в  $j$ -й час смены потребляет  $e_{ij}$  ( $0 \leq e_{ij} \leq 10^2$ ) условных единиц электроэнергии. Смена длится 8 часов. Если станок простаивает, то потребляемый им объём электроэнергии считается равным нулю. В электросеть цеха, в котором расположены станки, можно подать не более  $E$  ( $10 \leq E \leq 10^3$ ) условных единиц электроэнергии в час.

Можно считать, что эффективность смены пропорциональна суммарному объёму потреблённой станками электроэнергии, конечно, если соблюдается прописанный технологический процесс.

Генерального директора предприятия интересует, какая максимальная эффективность смены может быть достигнута производством при существующих ограничениях. Напишите программу, которая по заданным  $N$ ,  $E$  и  $e_{ij}$  найдёт интересующую величину.

### Входные данные

В первой строке через пробел указаны две величины  $E$  и  $N$ . В следующих  $N$  строках указаны через пробел по 8 чисел – потребляемые объёмы электроэнергии  $i$ -м станком, где  $i$  соответствует номеру строки, а номер числа в строке соответствует часу смены. Гарантируется, что все указанные во входных данных величины являются целыми числами.

### Выходные данные

Выведите одно целое число – интересующую величину.

### Примеры

Ввод	Вывод
3 2 1 2 3 2 4 1 2 3 3 2 1 4 0 0 1 0	0
4 3 1 2 3 2 4 1 2 3 3 2 1 4 0 0 1 0 0 1 1 2 0 2 1 0	25

## В. Склад

Ограничение по времени: **5000 миллисекунд**  
 Ограничение по памяти: **65000 кибибайт**

На большой склад областного снабжения поступает множество  $C$  ( $1 \leq C \leq 10^5$ ) товаров. Оптовые покупатели (предприниматели) стараются приобрести со склада самый дешёвый в закупе товар и покупают они на всю имеющуюся у них сумму денег.

Рассмотрим процесс приобретения товаров предпринимателем. Предприниматель приобретает на всю имеющуюся у него сумму  $D_k$  самые дешёвые товары. Затем он продаёт товары, и как только у него высвобождается сумма равная начальной, снова производит закуп на  $D_k$  средств. Заметьте, ассортимент и количество товара от закупа к закупу уменьшается. Описанный процесс повторяется  $N_k$  раз.

В данной задаче нужно подсчитать количество товаров, приобретённое предпринимателем за  $N$  операций покупки. Такой расчёт нужно произвести  $K$  раз для разных  $D_k$  и  $N_k$ .

### Входные данные

В первой строке указано одно число  $K$  ( $1 \leq K \leq 10^4$ ) – количество различных сумм и операций, для каждой из которых нужно найти ответ к задаче. Далее следует  $K$  пар чисел  $D_k$  ( $1 \leq D_k \leq 10^6$ ) и  $N_k$  ( $1 \leq N_k \leq 10^5$ ), где  $k$  ( $1 \leq k \leq K$ ) – порядковый номер пары  $D_k$  и  $N_k$ . Каждая пара чисел расположена на отдельной строке.

После на отдельной строке дано число  $C$  ( $1 \leq C \leq 10^5$ ) – количество различных товаров на складе. Далее в  $C$  строках даны описания товаров – каждое состоит из двух целых чисел  $price_c$  ( $1 \leq price_c \leq 10^4$ ) и  $count_c$  ( $1 \leq count_c \leq 10^5$ ), соответственно цена и количество имеющегося на складе товара, где  $c$  ( $1 \leq c \leq C$ ) – порядковый номер товара.

Гарантируется, что все исходные числа во входных данных целые.

### Выходные данные

Выведите  $K$  строк, на  $k$ -ой строке должно быть одно целое число – суммарное количество товаров, которые приобретёт предприниматель, если он будет действовать по указанному в задаче алгоритму и производить закуп  $N_k$  раз на сумму  $D_k$ .

### Примеры

Ввод	Вывод
2	3
5 1	4
4 3	
3	
3 2	
5 1	
1 2	

### Разъяснение примера

В первом случае у предпринимателя  $D_1 = 5$ ,  $N_1 = 1$ . При первом и единственном закупе предприниматель возьмёт два товара по цене 1 (см. 3-й товар) и один товар по цене 2 (см. первый товар). Средства закончились, количество закупов закончилось, следовательно, ответом для первого случая будет 3 (три).

Во втором случае у предпринимателя  $D_2 = 4$ ,  $N_2 = 3$ . При первом закупе будет приобретено два товара по цене 1 (см. 3-й товар) и останется 2-е условные единицы (у.е.) средств, больше ничего купить нельзя. Во втором закупе будет приобретён только 1 товар по цене 3 (см. 1-й товар) и останется 1 у.е., на неё уже в этом закупе ничего не купить. И в третьем, последнем (т.к.  $N_2 = 3$ ), закупе будет также приобретён один первый товар. Таким образом за три закупа во втором случае будет приобретено  $2 + 1 + 1 = 4$  товара.

## С. Чек

Ограничение по времени: **1000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

В терминалы, которые использует автобусный парк городского автотранспорта встроен принтер. Так как ширина ленты невелика, то желаемый текст следует разбивать (переносить) не взирая на правила переноса. Требуется написать программу, которая получает на входе текст сообщения в виде одной строки, состоящей не более чем из 1000

символов, а выводит этот же текст построчно, по  $k$  символов на строку (может быть только за исключением последней).

### Входные данные

В первой строке одно число  $k$  ( $8 \leq k \leq 64$ ) – количество символов на строку. Начиная с первого символа второй строки и до её конца – сообщение. Сообщение содержит только символы латиницы, цифры, пробелы, знаки пунктуации: точки, запятые, двоеточия, точки с запятыми, тире, восклицательные и вопросительные знаки, круглые открывающие и закрывающие скобки.

### Выходные данные

Сообщение разделённое по  $k$  знаков, каждую очередную порцию  $k$  знаков выводить с новой строки. Если входное сообщение начинается или заканчивается пробелами, они так же должны присутствовать и в выходных данных.

### Пример

Ввод	Вывод
8 Hello, World!	Hello, W orld!

## D. Заливка

Ограничение по времени: **1000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Заливка контура графической фигуры, графического примитива или произвольной формы является во всех современных графических редакторах обязательным инструментом. В этой задаче вам предложено реализовать любой способ заливки (последовательность не имеет значения). Заливка может распространяться от пикселя к пикселю, только если у них есть общая сторона. В качестве незакрашенного пикселя выступает знак точки «.», а в качестве покрашенного знак «#» (решётка, номер).

### Входные данные

В первой строке даны два целых числа  $X$ ,  $Y$  – координаты из области картины откуда начинается заливка, левый верхний угол имеет координаты  $(0, 0)$ . Дана картина размером  $N$  на  $M$  ( $1 \leq N, M \leq 20$ ).  $N$  – количество строк,  $M$  – количество столбцов.

### Выходные данные

Выведите результат заливки.

### Пример

Ввод	Вывод
1 1 .####. #..... #...## .####. .....	.##### ##### ##### .####. .....

Ввод	Вывод
5 3 .####. #..... #...## .####. .....	.####. #..... #...## ##### #####

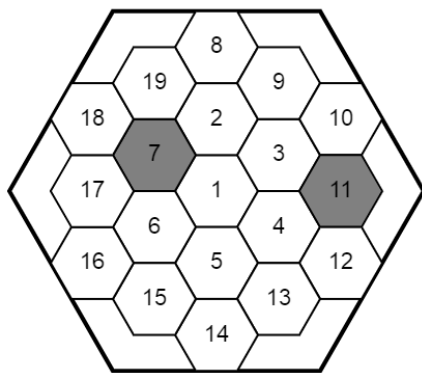
## Е. Расстояние

Ограничение по времени: **1000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Существуют различные требования к расстоянию между сотрудниками компании внутри одного помещения. Эти требования вызваны разными причинами, это требования гигиены, комфорта, эффективности работы, экономии места и др.

В одной из компаний по производству мёда на плане каждого помещения (в виде правильного шестиугольника) отображено замощение всей площади правильными шестиугольниками со стороной  $A$  (см. рисунок). Все шестиугольники внутри помещения нумеруются натуральными числами начиная с центрального. Далее по часовой стрелке начиная сверху (см. рисунок). На плане серой заливкой обозначены места расположения сотрудников компании во время работы (рабочие места сотрудников).



Требуется по имеющимся данным о рабочих местах определить текущее минимальное расстояние между рабочими местами сотрудников. Расстояние между двумя рабочими местами определяется как длина отрезка, соединяющего геометрические центры правильных шестиугольников.

### Входные данные

В первой строке дано единственное целое положительное число  $A$  ( $1 \leq A \leq 100$ ).

Далее в отдельной строке дано число  $K$  ( $2 \leq K \leq 19$ ) – количество рабочих мест. На следующей строке через пробел указано  $K$  чисел – номера шестиугольников  $n_k$  ( $1 \leq n_k \leq 19$ ), в которых расположены рабочие места.

### Выходные данные

Укажите одно число – минимальное расстояние между рабочими местами с точностью не хуже  $10^{-6}$ .

### Пример

Ввод	Вывод
3 2 7 11	13.74772708

## Ф. Пропеллеры

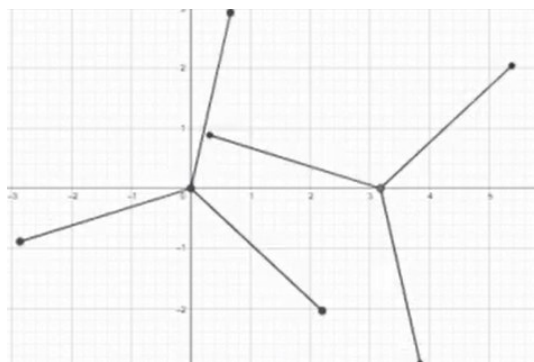
Ограничение по времени: **1000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Пропеллеры могут использоваться для получения воздушных направленных потоков с целью осуществления, например, охлаждения радиатора, нагреваемого выделяемым видеопроцессором теплом. С целью увеличения потока могут устанавливаться два и более воздушных винтов.

Для увеличения воздушного потока в компактных размерах видеокарты одна из компаний по производству систем охлаждения решила 2-а пропеллера разместить как можно ближе друг к другу. Они заметили, что центры пропеллеров можно разместить ближе друг к другу, если вращение организовать в разные стороны (по часовой и против часовой стрелки). Конечно, тогда нужно согласовать многие вещи. Инженеры уже научились выставлять начальное положение воздушных винтов (фазу) и обеспечивать им одинаковую скорость вращения.

Для предварительной оценки максимально возможного сближения центров трёхлопастных винтов вас просят написать программу, которая решит геометрическую задачу (см. рисунок). По известной длине лопасти  $D$  необходимо найти минимальное расстояние  $L$  между центрами винтов.



### Входные данные

В первой строке дано единственное целое число  $D$  ( $0 < D \leq 4 \times 10^3$ ).

### Выходные данные

Выведите  $L$  с точность не ниже  $10^{-6}$ .

### Пример

Ввод	Вывод
4	4.1921317

## Г. Среднее

Ограничение по времени: **2000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Скользящее среднее – часто используемый инструмент сглаживания данных краткосрочных колебаний интересующей величины, получаемых с датчиков. В данной задаче необходимо по имеющимся данным получить значения простой скользящей средней (арифметическое среднее).

Далее указана формула для подсчёта арифметического скользящего среднего.

Пусть дан ряд из  $N$  чисел, обозначим каждое число в ряду как  $n_i$ , где  $i$  – порядковый номер числа от 1 до  $N$  включительно. Пусть  $j$ -ое скользящее среднее определяется выражением  $\text{average}_j = (n_j + n_{j-1} + n_{j-2} + n_{j-3} + \dots + n_{j-k+1}) : k$ . При этом, если в последовательности для  $j$ -го среднего не хватает аргументов, то такое среднее не подсчитывается.

### Входные данные

В первой строке дано число  $N$  ( $1 \leq N \leq 10^6$ ) – количество значений в ряду и число  $K$  ( $1 \leq K \leq N$ ) – количество значений исходных последовательных данных для расчёта скользящего среднего (сглаживающий интервал). На следующей строке через пробел даны целые числа  $n_i$  ( $-10^6 \leq n_i \leq +10^6$ ) – значения ряда.

### Выходные данные

Через пробел последовательно с точностью не ниже  $10^{-6}$  данные скользящей средней от  $average_K$  до  $average_N$ .

### Пример

Ввод	Вывод
5 3 1 2 3 4 5	2 3 4
4 2 1 3 6 10	2 4.5 8

## Н. Шипение и гул

Ограничение по времени: **1000 миллисекунд**

Ограничение по памяти: **65000 кибибайт**

Для обучения детей правилам деления может быть использована такая групповая игра как *fizz buzz*. Начинаящий произносит число 1 (один), каждый следующий участник групповой игры увеличивает число на 1 (единицу) и если число делится на 3, произносит «fizz», если число делится на 5, произносит «buzz», если делится и на 3 и на 5, произносится «fizz buzz», в противном случае называется само полученное число.

Требуется по записанным данным игры указать количество совершённых ошибок. Под ошибкой понимается записанное не то, что должно было быть произнесено (не то число, не та фраза)!

### Входные данные

Строка ответов участников (не более  $10^5$  ответов). Каждый ответ отделён от другого запятой. Гарантируется, что все ответы – это либо натуральные числа, не превышающие  $10^5$ , либо слова «fizz», «buzz» или пара слов «fizz buzz».

### Выходные данные

Укажите одно целое неотрицательное число – количество ошибок, допущенных участниками групповой игры.

### Пример

Ввод	Вывод
1,2,fizz,3,buzz,fizz	1
1,2,fizz,4,buzz,fizz	0
1,2,fizz,4,buzz,fizz,7,8,fizz,buzz,11,fizz,13,14,fizz buzz,16,17,fizz,19,fizz,buzz	2