

## Задача А. Книги

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

На полке в ряд стоят  $N$  книг, по  $W$  листов каждая.

Книги пронумерованы слева направо начиная с единицы.

Книжный червь Костян решил ползти слева направо, сгрызая все листы на своём пути, пока не встретит первый лист  $k$ -й книги, который есть не собирается.

Сколько листов прогрызёт герой этой задачи?



### Формат входных данных

Первая строка содержит три целых числа  $N$ ,  $W$  и  $k$  ( $1 \leq N, W \leq 10^3$ ;  $1 \leq k \leq N$ ).

### Формат выходных данных

Выведите единственное число — количество листов, которые прогрызёт Костян.

### Пример

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
5 1 3	2

## Задача В. Рейд-Босс

Имя входного файла:	<i>стандартный ввод</i> или input.txt
Имя выходного файла:	<i>стандартный вывод</i> или output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В одной компьютерной RPG игре проходит инвент, на который собралось  $N$  игроков. Каждый из них имеет одного персонажа с уровнем  $L_i$ , являющимся целым числом.

Для того, чтобы начать бой с рейд-боссом, уровень каждого игрока должен являться точной  $D$ -степенью. Число  $A$  является точной  $D$ -степенью, если  $A = X^D$  для какого-то целого  $X$ .

К счастью, во внутриигровом магазине продаются зелья изменения уровней. Зелье объёма  $S$  стоит  $S$  монет, где  $S$  — любое целое положительное число. Игрок может применить зелье на персонажа и, по своему усмотрению, уровень станет либо в  $S$  раз больше, либо в  $S$  раз меньше. При этом второй случай допустим только если уровень персонажа на момент принятия зелья делится на  $S$ .

Зелье каждого объёма находится в неограниченном количестве. Игрок до начала боя может применить на персонажа сколько угодно зелий.

По заданным уровням игроков до начала рейда посчитайте минимальное количество монет, которое необходимо игрокам для допуска к бою.

### Формат входных данных

Первая строка содержит два целых числа  $N$  — количество игроков ( $1 \leq N \leq 10^5$ ), и  $D$  — требуемую степень ( $0 \leq D \leq 10^9$ ).

Во второй строке содержатся  $N$  целых чисел  $L_i$ , разделённых пробелом ( $1 \leq L_i \leq 10^6$ ).

### Формат выходных данных

Выведите единственное число — минимальную суммарную стоимость изменения уровней.

### Пример

стандартный ввод или input.txt	стандартный вывод или output.txt
5 2	10
1 2 3 4 5	



## Задача D. Стоунхендж 2077

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

*«Крэмлех — древнее сооружение, как правило, позднего неолита или раннего бронзового века, представляющее собой несколько поставленных вертикально в землю продолговатых камней, образующих одну или несколько концентрических окружностей.»*  
(Материал из Википедии — свободной энциклопедии)

В будущем 2077-м году миру открылся заповедник «Стоунхендж 2077» — продвинутая версия легендарной английской достопримечательности. Он представляет из себя  $n$  камней, расположенных по кругу и образующих правильный  $n$ -угольник. Камни пронумерованы от 1 до  $n$  по часовой стрелке, если смотреть на конструкцию сверху.

Начался туристический сезон и администрация заповедника хочет проводить экскурсии внутри заповедника. Каждая экскурсия будет соединять два различных камня. Между этими двумя камнями будет проложена прямая дорога, по которой будут ходить люди.

Ваша задача — организовать создание и закрытие экскурсии по запросам администрации. Главное правило, которое должно соблюдаться на протяжении всего времени: во избежание столкновений, никакие две экскурсии не должны пересекаться или иметь общий конец.

Формально, администрация будет давать вам команды двух типов:

1. «`link a b`» — запрос на создание экскурсии между камнями  $a$  и  $b$  ( $1 \leq a, b \leq n$ ;  $a \neq b$ ).

Если возможно создать такую экскурсию, то создать её и сообщить администрации её порядковый номер. Номера выдаются в порядке создания экскурсий, начиная с единицы.

Если же создание невозможно из-за пересечения с какими-либо уже существующими экскурсиями — сообщить об ошибке и не создавать.

2. «`cut k`» — запрос на отмену экскурсии с порядковым номером  $k$ .

Гарантируется, что на момент запроса экскурсия с таким номером существует в программе заповедника.

### Формат входных данных

В первой строке содержатся два целых числа  $n$  — количество камней, и  $m$  — количество команд ( $3 \leq n \leq 10^9$ ;  $1 \leq m \leq 3 \times 10^5$ ).

В следующих  $m$  строках содержатся команды в формате, описанном выше.

### Формат выходных данных

На каждую операцию первого типа выведите «`added as k`», где  $k$  — номер экскурсии, если её можно проводить в данный момент, и «`error`», если невозможно.

### Пример

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
10 8	added as 1
link 2 4	added as 2
link 9 5	error
link 3 8	error
link 4 5	added as 3
link 6 7	added as 4
cut 1	
cut 2	
link 3 8	

## Задача Е. Новый будильник

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Как-то раз Саша решил завести напоминание на своём телефоне. Он зашёл в приложение, отвечающее за время и обнаружил, что там уже есть  $N$  ранее установленных будильников. Его заинтересовал вопрос: за какое наименьшее количество действий он сможет выставить новый будильник?

Саша может совершать любое из следующих действий:

- Создать будильник на время `00:00`.
- Увеличить или уменьшить время в любом будильнике на один час. При уменьшении 0-го часа, он переходит в 23-й, и наоборот, при увеличении 23-го, он переходит в 0-й.
- Увеличить или уменьшить время в любом будильнике на одну минуту. Аналогично часам, ведут себя и 0-я и 59-я минуты.

Заметим, что изменяя минуты в будильнике, количество часов не меняется.

Определите за сколько действий Саша может завести будильник на требуемое время.

### Формат входных данных

В первой строке содержится целое число  $N$  — количество ранее установленных будильников ( $1 \leq N \leq 100$ ).

В каждой из следующих  $N$  строк содержится время очередного будильника.

В последней строке задано время напоминания, которое хочет установить Саша.

Каждое время задаётся в формате `hh:mm`. Минуты заданы от 0 до 59, часы от 0 до 23.

### Формат выходных данных

В единственной строке выведите минимальное количество действий.

### Пример

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
2 06:00 18:30 10:00	4

## Задача F. Тепло

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Вы сидите в холодной волшебной избе, в которой находится волшебная печь. Рядом с вами расположены  $n$  волшебных камней, на  $i$ -м камне высечено целое число  $a_i$ .

Камни можно кидать в печь, но особым образом — по два камня. Иначе произойдёт непоправимое. Не будем вдаваться в подробности, так написано в инструкции. После броска в печь камень нельзя использовать повторно.

После того как в печь будет кинута пара камней, на которых написаны числа  $x$  и  $y$ , к температуре в избе добавится  $x \times y$  градусов. Изначально температура равна нулю.

Определите, какие пары камней надо кидать в печь, чтобы в итоге температура в избе была максимально возможной.

### Формат входных данных

Первая строка содержит целое число  $n$  — количество камней ( $1 \leq n \leq 2 \times 10^5$ ).

Вторая строка содержит  $n$  целых чисел  $a_i$ , разделённых пробелами ( $|a_i| \leq 10^9$ ).

### Формат выходных данных

В первой строке выведите целое число  $m$  — количество пар камней в найденном вами способе получения максимальной температуры.

В следующих  $m$  строках выведите по два числа, написанных на очередной паре камней.

Если вариантов ответа несколько, разрешается вывести любой из них.

### Пример

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
4	2
17 16 19 8	17 19 16 8

## Задача G. Очень длинный корень

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Маленький Виталик только начал заниматься олимпиадным программированием. Так как он ещё не знает алгоритмов, то решил начать с задач, в которых они не требуются.

Сегодня ему попала задача, в которой нужно вычислить целочисленный корень из длинного целого числа  $N$ , то есть необходимо найти максимальное целое  $X$  такое, что  $X^2 \leq N$ .

Виталик уже приступил к реализации своего решения, но тут у него возникла проблема. Он хочет хранить результат в массиве, в котором каждый элемент соответствует десятичному разряду в числе, при этом длина этого массива должна быть минимальна (для экономии памяти).

Виталик понял, что не умеет вычислять размер массива, в который запишет ответ, и просит вас о помощи.

### Формат входных данных

В первой строке задано одно целое число  $N$  ( $1 \leq N \leq 10^{100000}$ ).

### Формат выходных данных

В единственную строку выведите длину требуемого массива.

### Пример

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
36	1

## Задача Н. Плохой хеш

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Вам знакомо понятие хеширования? Это способ представления какого-либо объекта в виде целого числа — хеша. Одинаковые объекты должны иметь одинаковый хеш, а вот разные не обязательно: их хеш может совпадать. Но чем чаще такое происходит, тем хуже метод хеширования.

Маша придумала свой способ хеширования целых чисел: за хеш числа  $X$  она принимает  $S(X)$  — сумму всех циклических сдвигов числа  $X$ .

Например,  $S(47) = 47 + 74 = 121$ , а  $S(9090) = 9090 + 909 + 9090 + 909 = 19998$ .

Ваша задача — взломать хеш: для данного  $H$  найти количество таких  $X$ , что  $S(X) = H$ .

### Формат входных данных

Первая строка содержит целое число  $T$  — количество тестов ( $1 \leq T \leq 50$ ).

Следующие  $T$  строк содержат по одному целому числу  $H$  ( $1 \leq H \leq 10^{18}$ ).

### Формат выходных данных

На каждый тест выведите ответ в отдельной строке — количество способов раскодировать число.

### Пример

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
3	1
7	2
22	0
123	



## Задача I. олсич еоньламискаМ

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Думаете, числам не могут сниться сны? Ещё как могут, да и к тому же коллективные!

Вот и сейчас целым числам с отрезка  $[L, R]$  снится, как они попали в зазеркалье. При этом каждое число в таком сне становится перевёрнутым. Например, 82736 становится 63728, а 17740 становится 4771.

Чтобы выйти из сна, числам нужно преодолеть несколько испытаний, но в начале необходимо определиться с лидером — числом, которое в зазеркалье является наибольшим.

Окажите услугу, определите это число как можно скорее.

### Формат входных данных

Ввод содержит два целых числа: в первой строке число  $L$ , во второй —  $R$  ( $0 < L \leq R < 10^{100}$ ).

### Формат выходных данных

В единственную строку выведите максимальное число без лидирующих нулей.

### Примеры

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
1 10	9
5 15	51

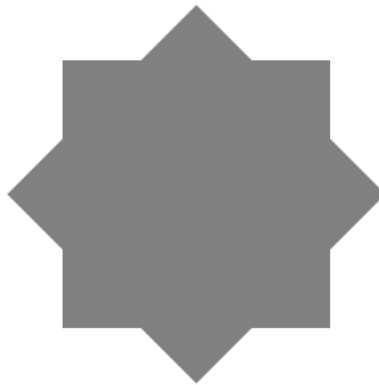
## Задача J. Дроны фермера Джона

Имя входного файла: *стандартный ввод* или `input.txt`  
Имя выходного файла: *стандартный вывод* или `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Фермер Джон решил освоить современные технологии и приобрёл двух дронов-поливальщиков. Дрон может поливать участок поля, являющийся квадратом. Первому дрону Джон установил полить удобрениями квадрат со стороной  $A$ , второму — со стороной  $B$ .

Джон запустил дронов из одной точки, и пошёл отдыхать к себе на веранду, попивая прохладный лимонад. «Новые технологии действительно облегчают жизнь»,- подумал Джон.

Но что-то пошло не так, и второй дрон полил участок не там, где ему было указано. Он полил квадрат, центр которого совпадает с центром участка первого дрона, и при этом второй квадрат повернут на  $45$  градусов относительно первого.



политая обоими дронами область для первого примера

Теперь Джону интересно, всё ли так плохо, как кажется? Посчитайте площадь поля, удобренную хотя бы одним дроном.

### Формат входных данных

Единственная строка содержит два целых числа  $A$  и  $B$  ( $1 \leq A, B \leq 10^4$ ).

### Формат выходных данных

Выведите площадь, политую обоими дронами.

Выведенный вами ответ будет считаться верным, если абсолютная или относительная погрешность не превосходит  $10^{-6}$ .

### Примеры

стандартный ввод или <code>input.txt</code>	стандартный вывод или <code>output.txt</code>
2 2	4.6862915
3 1	9