

## Задача А. Поворот на МEGУ

Имя входного файла: `canteen.in`  
Имя выходного файла: `canteen.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Светофор, установленный на повороте в торговый центр «МЕГА» с Новомосковского тракта, работает в таком режиме, что за одну минуту повернуть успевают  $k$  машин. По выходным, когда жители города едут закупаться продуктами и одеждой, перед этим поворотом вырастает огромная пробка. Администрация торгового центра дала указание повесить под строящимся неподалёку мостом камеру, подсчитывающую количество автомобилей, подъезжающих к этому повороту со стороны города. По информации, полученной с камеры за  $n$  минут, прошедших с начала наблюдений, определите количество машин, стоящих в пробке в настоящий момент времени.

### Формат входного файла

В первой строке записаны целые числа  $k$  и  $n$  ( $1 \leq k, n \leq 100$ ) — количество машин, успевающих повернуть на «МЕГУ» в течение минуты, и количество минут, прошедших с начала наблюдений. Во второй строке через пробел записаны целые числа  $a_1, \dots, a_n$  ( $0 \leq a_i \leq 100$ ), где  $a_i$  — количество машин, подъехавших к повороту со стороны города в течение  $i$ -й минуты. Можно считать, что наблюдения начинаются рано утром, когда машин, ожидающих на повороте, ещё нет.

### Формат выходного файла

Выведите количество машин, стоящих в пробке на повороте через  $n$  минут после начала наблюдений.

### Примеры

<code>canteen.in</code>	<code>canteen.out</code>
5 3 6 7 2	0
5 3 20 0 0	5

## Задача В. Шедевры мировой архитектуры

Имя входного файла: `game.in`  
Имя выходного файла: `game.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Мария Ивановна оповестила всех учеников своего пятого класса, что через месяц она проведёт урок на тему «Шедевры мировой архитектуры». К этому уроку каждый из ребят должен был подготовить небольшой доклад об одном из знаменитых архитектурных сооружений. Как обычно, отличники подготовили свой доклад заранее, а двоечники начали готовиться к уроку только за несколько минут до него.

Начался урок. По сложившейся традиции проведения подобных уроков, дети садятся в круг и выступают один за другим по часовой стрелке. Отличники любят выступать одними из первых, в то время как двоечники хотят отвечать как можно позже, пытаются доделать задание прямо во время урока.

Мария Ивановна спросила у каждого ученика, каким по счёту он хочет выступать. Теперь ей предстоит решить, кто должен выступить первым, чтобы как можно больше детей выступили по счёту ровно теми, какими они хотят.

### Формат входного файла

В первой строке записано целое число  $n$  — количество учеников в классе ( $2 \leq n \leq 10^5$ ). Мария Ивановна занумеровала всех детей числами от 1 до  $n$  в порядке по часовой стрелке. Во второй строке через пробел записаны числа  $a_1, \dots, a_n$  ( $1 \leq a_i \leq n$ ), где  $a_i$  — номер, названный  $i$ -м учеником.

### Формат выходного файла

Выведите номер ученика, которого Мария Ивановна спросит первым. Если возможно несколько вариантов ответа, выведите любой.

### Примеры

<code>game.in</code>	<code>game.out</code>
4 4 1 2 3	2
3 1 1 1	3

## Задача С. Код Хэмминга

Имя входного файла: `hamming.in`  
Имя выходного файла: `hamming.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Рассмотрим четыре круга, пересекающиеся так, как показано на рисунке. Назовём *лепестком* каждую из трёх фигур, образованных пересечением трёх кругов.

Запишем в каждом из кругов ноль или единицу. После этого в каждом лепестке запишем остаток при делении на два суммы чисел во всех кругах, в которых содержится этот лепесток. Например, если в кругах были записаны числа 0, 1, 0, 1, то в лепестках будут записаны числа 0, 1, 0 (круги и лепестки перечислены в порядке, указанном на рисунке).

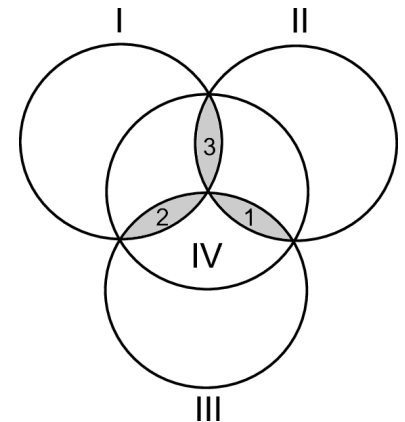
Описанная схема называется *кодом Хэмминга* и обладает интересным свойством. Если ваш враг в тайне от вас изменит любое из семи записанных по этой схеме чисел, вы сможете однозначно определить, какое число он изменил. Решив эту задачу, вы узнаете, как это сделать.

### Формат входного файла

В единственной строке через пробел записаны семь чисел. Каждое из чисел равно нулю или единице. Сперва идут четыре числа, записанные в кругах в порядке, указанном на рисунке. Далее идут три числа, записанные в лепестках в порядке, указанном на рисунке.

### Формат выходного файла

В единственной строке выведите через пробел семь чисел, образующие код Хэмминга. Набор чисел должен отличаться от исходного не более чем в одном числе. Гарантируется, что любой набор входных данных либо сам является кодом Хэмминга, либо в нём можно изменить в точности одну цифру и получить код Хэмминга.



## Примеры

hamming.in	hamming.out
0 1 0 1 1 0 1	0 1 0 0 1 0 1
1 1 1 1 1 1 1	1 1 1 1 1 1 1

## Задача D. Трудности локализации

Имя входного файла: `heroes.in`  
Имя выходного файла: `heroes.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Компания Lavin Interactive, разработчик пошаговой стратегии Losers-V, постоянно расширяет рынки сбыта и создаёт локализации своей игры даже на самые малоизвестные языки. В том числе, их заинтересовал язык племени австралийских аборигенов аиндилъяква.

Но в языке аиндилъяква нет числительных. Как же, например, перевести на него фразу «у вас семь чёрных драконов, а у вашего врага — сорок»? Локализаторы решили перевести её так: «у вас немного чёрных драконов, а у вашего врага — толпа». Они составили таблицу, в которой указали правила замены чисел, обозначающих количество монстров, на существительные и местоимения:

Количество	Обозначение на русском языке	Обозначение на языке аиндилъяква
от 1 до 4	несколько	few
от 5 до 9	немного	several
от 10 до 19	отряд	pack
от 20 до 49	толпа	lots
от 50 до 99	орда	horde
от 100 до 249	множество	throng
от 250 до 499	сонмище	swarm
от 500 до 999	полчище	zounds
от 1000	легион	legion

Помогите локализаторам автоматизировать процесс — напишите программу, которая по количеству монстров выдаст соответствующее этому количеству слово.

### Формат входного файла

В единственной строке записано целое число  $n$  ( $1 \leq n \leq 2000$ ) — количество монстров.

### Формат выходного файла

Выведите слово, соответствующее данному количеству монстров на языке аиндилъяква.

## Примеры

heroes.in	heroes.out
7	several
40	lots

## Задача E. Биография Сандро

Имя входного файла: `legend.in`  
Имя выходного файла: `legend.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Леогиус искал в библиотеке книгу, которую ему посоветовал учитель теоретической магии. Вдруг он наткнулся на древнюю летопись, записанную на нескольких листах пергамента. Бегло проглядев её, Леогиус понял, что она описывает жизнь и удивительные приключения одного лица. Неужели этот манускрипт — биография лица Сандро, утерянная много веков назад? Если это так, находку нужно как можно быстрее показать Верховному Совету Магов. Одна беда — в тексте летописи ни разу не упоминается имя Сандро. Что же делать? Совет может и не поверить, что летопись повествует именно о Сандро.

Леогиус решил, что летопись нужно срочно исправить. Довольно быстро нашёлся маг, готовый сделать это. Но за хорошую работу всегда нужно платить. Корректор был готов исправить любую букву на любую другую букву того же регистра (заглавную — на заглавную, строчную — на строчную) за пять золотых. Кроме того, он мог изменить регистр любой буквы, также за пять золотых. Помогите Леогиусу определить, какое минимальное количество золотых ему придётся заплатить, чтобы в тексте появилась строка `Sandro`.

### Формат входного файла

Единственная строка содержит текст летописи, состоящий из строчных и заглавных латинских букв. В тексте не менее шести и не более 200 символов.

### Формат выходного файла

Выведите минимальное количество золотых, которое придётся заплатить, чтобы в тексте появилось имя Сандро.

## Пример

legend.in	legend.out
MyNameIsAlexander	20

В примере корректору придётся выполнить четыре операции, после которых строка последовательно будет принимать следующий вид: `MyNameIsAlesander`, `MyNameIsAlesandrr`, `MyNameIsAlesandro`, `MyNameIsAleSandro`.

## Задача F. В поисках Додекаэдра

Имя входного файла: `mouse.in`  
Имя выходного файла: `mouse.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Великий Додекаэдр — древний и могучий артефакт. Как и другие магические многогранники, он хранится в Храме Пяти Полиэдров. Многие маги пытались заполучить его, но их старания не увенчались успехом. Всё дело в том, что на Додекаэдр наложено охранное заклинание.

В четвёртом зале Храма стоит ряд из  $n$  постаментов, пронумерованных слева направо, начиная с единицы. На одном из постаментов располагается сам Великий Додекаэдр, а на всех остальных — его точные копии, не обладающие магической силой.

Если маг прикоснётся к настоящему Додекаэдру, все его копии тут же исчезнут. Если же он дотронется до копии, то ничего не произойдёт, но как только маг уберёт руку, Додекаэдр переместится на один постамент влево или вправо, а на его месте появится копия.

Разумеется, заклинание пресекает все попытки добыть Додекаэдр нечестным путём — одновременное прикосновение к нескольким додекаэдрам приведёт к гибели мага.

Вот уже много веков маги-теоретики всего мира пытаются придумать алгоритм обнаружения Великого Додекаэдра. Пока им это не удалось. Может быть, вам удастся им помочь?

### Формат входного файла

В единственной строке записано целое число  $n$  ( $2 \leq n \leq 100$ ) — количество постаментов в четвёртом зале Храма Пяти Полиэдров.

### Формат выходного файла

В первой строке выведите целое число  $m$  ( $m \leq 1000$ ) — количество прикосновений, необходимых для обнаружения Додекаэдра. Во второй строке через пробел выведите  $m$  чисел — номера постаментов в том порядке, в котором нужно прикасаться к установленным на них додекаэдрам.

Алгоритм должен обнаруживать артефакт при любом его начальном положении и вне зависимости от его перемещений. Гарантируется, что существует хотя бы один такой алгоритм, в котором требуется совершить не более тысячи прикосновений.

### Пример

mouse.in	mouse.out
3	2 2 2

## Задача G. В поисках Истины

Имя входного файла: mousecheck.in  
Имя выходного файла: mousecheck.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Уже не первый день Верховный Совет Магов изучает найденную недавно биографию лица Сандро. Сейчас члены Совета исследуют таинственную последовательность чисел, записанную на одной из страниц. Большинство магов считает, что эта последовательность задаёт алгоритм обнаружения Великого Додекаэдра.

Как вы помните, Великий Додекаэдр хранится в Храме Пяти Полиэдров. Многие маги пытались заполучить его, но их старания не увенчались успехом — на Додекаэдр наложено охранное заклинание.

В четвёртом зале Храма стоит ряд из  $n$  постаментов, пронумерованных слева направо, начиная с единицы. На одном из постаментов располагается сам Великий Додекаэдр, а на всех остальных — его точные копии, не обладающие магической силой.

Если маг прикоснётся к настоящему Додекаэдру, все его копии тут же исчезнут. Если же он дотронется до копии, то ничего не произойдёт, но как только маг уберёт руку, Додекаэдр переместится на один постамент влево или вправо, а на его месте появится копия.

Верховный Совет Магов полагает, что числа в последовательности соответствуют номерам постаментов и показывают, в каком порядке нужно прикасаться к додекаэдрам. Помогите Совету проверить, что такой алгоритм позволит обнаружить артефакт при любом его начальном положении и вне зависимости от его перемещений

### Формат входного файла

В первой строке через пробел записаны целые числа  $n$  и  $m$  ( $2 \leq n, m \leq 100000$ ) — количество постаментов и количество чисел в последовательности. Во второй строке записана сама последовательность —  $m$  целых положительных чисел, не превосходящих  $n$ .

### Формат выходного файла

Выведите YES, если последовательность задаёт верный алгоритм обнаружения Великого Додекаэдра, и NO в противном случае.

### Примеры

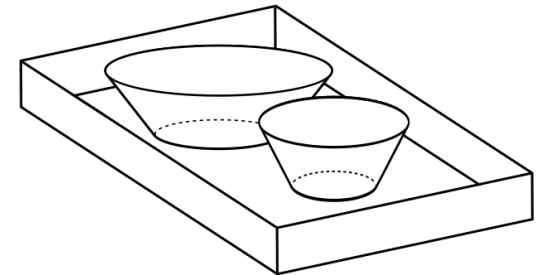
mousecheck.in	mousecheck.out
3 2 2 2	YES
3 3 1 2 3	NO

## Задача H. Тарелки 2

Имя входного файла: plates.in  
Имя выходного файла: plates.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Однажды один из организаторов Уральской региональной командной олимпиады по программированию зашёл в столовую перекусить. Взял суп и второе и попытался разместить их на небольшом прямоугольном подносе. Это оказалось не так уж просто. «Да уж, незадача,— пробормотал он. — Хотя, почему незадача? Как раз задача! Прямо-таки замечательная задача для олимпиады!»

У подносов в столовой УрГУ прямоугольное дно размером  $a \times b$  и вертикальные бортики высоты  $d$ . Все тарелки имеют форму усечённых конусов высоты  $h$ . Организатор хочет поставить тарелки на поднос так, чтобы их дно полностью прилегало к дну подноса. Удастся ли ему это сделать?



### Формат входного файла

В первой строке через пробел записаны целые числа  $a$ ,  $b$  и  $d$ . Каждая из следующих двух строк описывает одну из тарелок и содержит два целых числа. Первое число обозначает радиус дна тарелки, а второе — радиус окружности края тарелки. Радиус окружности края больше радиуса окружности дна. В последней строке записано целое число  $h$  — высота тарелок. Все числа во входных данных положительные и не превосходят 1000.

### Формат выходного файла

Выведите YES, если тарелки можно разместить на подносе, и NO в противном случае.

## Примеры

plates.in	plates.out
10 10 10 1 2 1 2 5	YES
8 4 1 1 2 1 3 1	NO

## Задача I. Мужья в магазине

Имя входного файла: shop.in  
Имя выходного файла: shop.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Марк вернулся домой после тяжёлого рабочего дня. Жена вместо того, чтобы накормить ужином, отправила его за хлебом. Когда Марк зашёл в ближайший магазин, он увидел длинную очередь из таких же несчастных мужей, как и он. Марк встал в конец очереди. К счастью, очередь двигалась достаточно быстро — каждому из этих мужей дома заказали только один вид товара.

Когда очередной муж подходит к прилавку и узнаёт у продавщицы, что в наличии есть нужное ему количество товара, он покупает его и уходит. Если продавщица отвечает ему, что такого товара совсем не осталось, то он сильно расстраивается и уходит домой с пустыми руками. Если же такого товара осталось меньше, чем нужно. . . Тут не обойтись без консультации жены. Чтобы не задерживать очередь, покупатель пропускает вперёд одного человека, вставая непосредственно за ним, и звонит жене. Жена всегда советует купить весь остаток товара. После этого он снова подходит к прилавку и, если количество товара, имеющееся в наличии, не изменилось, покупает его, иначе он снова пропускает вперёд одного человека и звонит жене. При этом если после очередного звонка продавщица говорит, что товар закончился, покупатель выходит из очереди и отправляется домой ни с чем.

Марк очень хочет вернуться домой и наконец поужинать. Определите, сколько минут пройдёт перед тем, как он впервые окажется у прилавка. Один покупатель проводит у прилавка ровно одну минуту, телефонный разговор с женой длится полминуты.

## Формат входного файла

В первой строке записано целое число  $m$  ( $1 \leq m \leq 1000$ ) — количество видов товаров, которыми торгует магазин. В каждой из следующих  $m$  строк описывается один товар. Описание товара имеет вид *amount of name*, где *name* — название товара (непустая строка из строчных латинских букв длиной не более 20), а *amount* — количество единиц товара, которое имелось в наличии в тот момент, когда Марк зашёл в магазин (положительное целое число, не превосходящее 1000). Названия всех товаров различны.

В следующей строке записано целое число  $n$  ( $1 \leq n \leq 1000$ ) — количество покупателей в очереди перед Марком. В каждой из следующих  $n$  строк описывается товар, который хочет купить очередной покупатель, в том же формате, в котором описаны товары в магазине. Покупатель мог прийти в магазин за товаром, который в этом магазине вообще не продаётся. Информация о покупателях дана в порядке от начала к концу очереди.

## Формат выходного файла

Выведите, сколько минут Марк простоит в очереди перед тем, как впервые подойдёт к прилавку.

## Пример

shop.in	shop.out
3 2 of sweets 4 of milk 1 of sausage 4 2 of milk 3 of sweets 3 of milk 1 of cheese	6

## Задача J. Парк аттракционов

Имя входного файла: tickets.in  
Имя выходного файла: tickets.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

В солнечное воскресенье группа детей во главе с учителем отправилась в парк аттракционов. В тот день на кассе в парке сидела очень добрая и спокойная тётя Фрося. Учитель отдал ей деньги, но не сказал, сколько билетов ему нужно. Может ли тётя Фрося определить это только по набору купюр, которые дал ей учитель? Она считает, что учитель не давал лишних купюр, то есть, если убрать из набора любую купюру, то денег на все билеты уже не хватит.

## Формат входного файла

В первой строке через пробел записаны шесть неотрицательных целых чисел — количества купюр достоинством 10, 50, 100, 500, 1000 и 5000 рублей, соответственно, которыми учитель расплатился с тётей Фросей. Во второй строке записано целое положительное число — цена одного билета. Все числа во входных данных не превосходят 1000.

## Формат выходного файла

Определите, сколько билетов хотел купить учитель. В первой строке выведите количество вариантов ответа, а во второй — все эти варианты через пробел, упорядоченные по возрастанию. Гарантируется, что хотя бы один вариант ответа существует.

## Примеры

tickets.in	tickets.out
0 2 0 0 0 0 10	5 6 7 8 9 10
1 2 0 0 0 0 10	1 11

## Задача К. О пользе зонтов

Имя входного файла: `umbrellas.in`  
Имя выходного файла: `umbrellas.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Группа выпускников одиннадцатого класса решила отпраздновать свой выпускной в аквапарке. Ребята отлично провели там время, но по выходу из аквапарка их ждал неприятный сюрприз — на улице резко похолодало и пошёл сильный дождь. Что же делать, неужели им всем придётся мокнуть по пути на троллейбусную остановку?

Тут выяснилось, что все юноши проявили предусмотрительность и взяли с собой зонты, в то время как ни у одной из девушек зонта с собой не оказалось. Конечно, каждый юноша, как настоящий джентльмен, вынужден был провести одну из девушек до троллейбусной остановки под своим зонтом.

Если  $i$ -й девушке придётся мокнуть под дождём, то она расстроится на величину  $g_i$ . Если же ни одна девушка не примет приглашение  $j$ -го юноши, то он расстроится на величину  $b_j \cdot k$ , где  $k$  — количество более удачливых юношей, которые будут сопровождать девушек под своим зонтом. Девушки, которые будут идти под зонтом, и юноши, которые их будут сопровождать, не расстроятся совсем.

Помогите ребятам сделать так, чтобы праздник был испорчен как можно меньше — определите, каким образом они должны идти на остановку, чтобы суммарное расстройство было минимальным.

### Формат входного файла

В первой строке записаны целые числа  $n$  и  $m$  ( $1 \leq n, m \leq 100$ ) — количество девушек и юношей в группе, соответственно. Во второй строке через пробел записаны числа  $g_1, \dots, g_n$  — расстройств девушек. В третьей строке через пробел записаны числа  $b_1, \dots, b_m$  — коэффициенты расстройств юношей. Все коэффициенты целые, положительные и не превосходят 1000.

### Формат выходного файла

Выведите единственное целое число — минимально возможное суммарное расстройство.

### Пример

<code>umbrellas.in</code>	<code>umbrellas.out</code>
2 4 1 100 10 8 6 4	19

## Задача Л. Дядя Стёпа и автобусы

Имя входного файла: `wicket.in`  
Имя выходного файла: `wicket.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Правительство области выделило деньги на ремонт участка автодороги Большие Васюки — Малые Васюки. Дорога на этом участке двухполосная, поэтому решено было сначала закрыть на ремонт одну полосу, оставив другую для движения транспорта в обоих направлениях.

Машинам, следующим в разных направлениях, пришлось проезжать этот участок по очереди. Естественно, с обеих сторон от него быстро скопились автомобильные пробки. Милиционеру Дяде Стёпе поручили регулировать дорожное движение на ремонтируемом участке дороги. Дядя Стёпа не растерялся, тут же нашёл пару неплохих просёлочных дорог и расставил в нужных местах указатели «Объезд». Казалось бы, проблема была решена...

Но оказалось, что по просёлочным дорогам не смогут проехать автобусы, курсирующие между Большими и Малыми Васюками. К счастью, автобусы ездят строго по расписанию, а значит, Дядя Стёпа заранее знает, когда и с какой стороны приедет очередной автобус. Также для каждого автобуса известно максимальное время, которое он сможет потратить на преодоление ремонтируемого участка, не опоздав при этом в пункт назначения. Дядя Стёпа передал эти данные вам и попросил написать программу, которая поможет ему организовать процесс таким образом, чтобы все автобусы прибыли на место вовремя.

Автобус может выехать на единственную свободную полосу только в том случае, если на ней нет других автобусов. Любой автобус проезжает ремонтируемый участок ровно за одну минуту.

### Формат входного файла

В первой строке записано целое число  $n$  ( $1 \leq n \leq 100\,000$ ) — количество автобусов, следующих из Малых Васюков в Большие. Следующие  $n$  строк содержат описание этих автобусов в виде пар целых чисел  $t_i$  и  $p_i$  ( $1 \leq t_i, p_i \leq 10^8$ ), где  $t_i$  — момент времени в минутах, в который ожидается прибытие  $i$ -го автобуса, а  $p_i$  — максимальное количество минут, которое этот автобус может потратить на преодоление ремонтируемого участка. В следующей строке записано целое число  $m$  ( $1 \leq m \leq 100\,000$ ) — количество автобусов, следующих из Больших Васюков в Малые. Следующие  $m$  строк содержат их описание в том же формате.

Автобусы описываются в том порядке, в котором они будут подъезжать к ремонтируемому участку. Автобусы, следующие в одном направлении, должны преодолевать этот участок в этом же порядке.

### Формат выходного файла

Выведите YES, если Дядя Стёпа сможет организовать движение так, чтобы ни один автобус не опоздал, и NO в противном случае.

### Примеры

<code>wicket.in</code>	<code>wicket.out</code>
2 1 1 1 2 1 2 2	YES
2 1 1 1 2 1 2 1	NO