

Задача А. Театр начинается с вешалки?

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В Омском драматическом театре расширили гардероб. Нужно сделать номерки на новые вешалки.

Посчитайте – сколько новых номерков нужно сделать.

Формат входных данных

В единственной строке через пробел находятся два числа a и b ($1 \leq a < b \leq 1000$).
 a – первый номер новых вешалок, b – последний номер новых вешалок.

Формат выходных данных

Вывести одно число – сколько новых номерков необходимо сделать.

Примеры

Ввод	Вывод
1 6	6
21 45	25

Задача В. Пила «Unity»

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вася очень любит своего деда, ведь он каждое лето проводил в деревне, пока был маленький. Недавно Василий устроился н работу менеджером и с первой зарплаты решил сделать деду подарок.

Рассмотрев разные варианты, Василий остановил свой выбор на электропиле «Unity». Купив пилу, счастливый Вася помчался к деду и решил показать, как теперь здорово пилить бревна для дров на зиму. Вася пилил и радовался, а дед радовался, что не ему пилить.

Постепенно Василий стал уставать и решил сделать перерыв. Пока он сидел и отдыхал задумался... Ведь можно было с самого начала понять сколько предстоит проделать распилов бревен, если знаешь сколько бревен и на сколько чурок нужно пилить каждое бревно. Однако он уже был достаточно уставший и не смог выполнить эту работу поэтому обратился за помощью к Вам.

Требуется написать программу, которая зная сколько бревен есть в наличие и насколько чурок нужно распилить каждое бревно определяет сколько всего распилов нужно сделать.

Формат входных данных

В первой строке через пробел расположено два числа k и p ($1 \leq k, p \leq 1000$), где k – количество бревен, которые нужно распилить, а p – на сколько чурок нужно распилить каждое бревно.

Формат выходных данных

Необходимо вывести одно число – количество распилов, которые необходимо сделать.

Примеры

<i>Ввод</i>	<i>Вывод</i>
1 3	2

Задача С. Коробки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

На заводе по производству детских игрушек в смену производится A шаров и B кубиков. Шары и кубики необходимо упаковывать в разные коробки.

Известно, что в коробку для шаров помещается P шаров, а в коробку для кубиков помещается Q кубиков.

Для оптимизации работы необходимо посчитать какое количество коробок для шаров и какое количество коробок для кубиков необходимо подготовить к каждой смене, при условии, что все шары и кубики должны быть упакованы, даже если коробки не полные.

Формат входных данных

В единственной строке через пробел расположены 4 натуральных числа – A , B , P и Q ($1 \leq A, B, P, Q \leq 1000$).

Формат выходных данных

Необходимо вывести через пробел 2 числа – количество коробок для упаковки шаров и количество коробок для упаковки кубиков.

Примеры

<i>Ввод</i>	<i>Вывод</i>
4 6 2 3	2 2
9 14 5 6	2 3

Задача D. Башня

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Аркадий, младший брат Васи строит башни из кубиков. Основанием башни является прямоугольник $N \times M$ кубиков. Башня строится слой за слоем. Пока слой полностью не заполнен, Аркадий не строит новый слой башни.

Наблюдая со стороны, Вася задумался, а можно ли посчитать, сколько полностью законченных слоев башни размером $N \times M$ можно сложить, если известно, сколько кубиков есть в наличии.

Сам Вася справится с такой задачей не в силах, поэтому он просит ответ у Вас.

Формат входных данных

В первой строке через пробел расположено 3 натуральных числа – N , M , K ($1 \leq N, M, K \leq 1000$), где K – количество кубиков, из которых нужно сложить башню, в которой каждый слой размером $N \times M$.

Формат выходных данных

Необходимо вывести одно число – количество законченных слоев башни.

Примеры

<i>Ввод</i>	<i>Вывод</i>
2 2 3	0
2 2 8	2

Задача Е. Маршрутки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В час пик на остановку одновременно подъехали три маршрутных такси, следующие по одному маршруту, в которые тут же набились пассажиры. Водители обнаружили, что количество пассажиров в разных маршрутках разное, и решили пересадить часть пассажиров так, чтобы в каждой маршрутке было поровну пассажиров. Требуется определить, какое наименьшее количество пассажиров придется при этом пересадить.

Формат входных данных

В единственной строке через пробел расположены три числа N , M , K ($1 \leq N, M, K \leq 100$).

Формат выходных данных

Необходимо вывести одно число – наименьшее количество пассажиров, которое требуется пересадить.

Если это сделать невозможно, вывести слово IMPOSSIBLE (заглавными буквами).

Примеры

<i>Ввод</i>	<i>Вывод</i>
1 2 3	1
99 100 100	IMPOSSIBLE

Задача F. Бассейн

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Бригада отделочников получила заказ: обклеить бассейн размером $N \times M \times H$ кафельной плиткой размером $q \times w$.

Прораб Петров, недавно окончил строительный университет и пытается в своей работе использовать передовые строительные, а также информационные технологии, поэтому он хочет при помощи компьютера рассчитать, какое минимальное количество плитки ему понадобится для выполнения данного задания. Однако в университете Петров часто прогуливал уроки по информатике поэтому единственные кто может ему помочь – это Вы.

Требуется по входным данным бассейна $N \times M \times H$, где N – длина бассейна, M – ширина, а H – глубина и кафельной плитки $q \times w$ (все числа натуральные не превосходят 100) определить количество плиток, которое потребуется для оклейки бассейна.

Примечание:

Если в процессе оклейки бассейна некоторые плитки приходится обрезать, то оставшиеся части плитки использовать повторно нельзя!

Во время оклейки, плитки, располагающиеся на одной стене или дне должны быть ориентированы одинаково.

Формат входных данных

В первой строке через пробел располагаются три числа N , M и H , во второй строке через пробел расположены два числа q и w ($1 \leq N, M, H, q, w \leq 100$).

Формат выходных данных

Выведите единственное число – количество плиток, необходимых для оклейки бассейна.

Примеры

Ввод	Вывод
3 5 2 1 1	47
2 2 3 2 2	9
2 3 4 1 2	23

Задача G. Счастливые числа

Имя входного файла: стандартный ввод

Имя выходного файла: стандартный вывод

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Разные люди придумывают числам разные свойства. Например, Вася придумал, что счастливыми числами называются числа, состоящие только из цифр «4» и «7». Найдите количество счастливых чисел, не превышающих n .

Формат входных данных

В единственной строке расположено единственное число n ($1 \leq n \leq 10^6$).

Формат выходных данных

Выведите количество счастливых чисел, не превышающих n .

Примеры

<i>Ввод</i>	<i>Вывод</i>
9	2
4 4	3

Задача Н. Печеньки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Аркадий – очень любит печеньки, но при этом он очень справедливый и когда мама покупает пачку печенек Аркадий делит печеньки поровну себе, маме и папе, а остаток забирает себе за работу.

В какой-то момент Аркадий решил посчитать – сколько печенек он съел за последние дни. Сам он пока считает медленно, а дней прошло много, помогите Аркадию сосчитать количество съеденных печенек.

Формат входных данных

В первой строке расположено одно число n — количество пачек печенек, купленных за ($1 \leq n \leq 1000$).

Во второй строке расположено n чисел a_1, \dots, a_n — количество печенек в i -ой пачке ($1 \leq a_i \leq 30$).

Формат выходных данных

Выведите одно число – количество печенек, которые съел Аркадий.

Примеры

<i>Ввод</i>	<i>Вывод</i>
3 12 10 13	13
7 2 2 3 3 4 6 5	13

Система оценки

Тесты к этой задаче состоят из нескольких групп:

- Группа 0 (0 баллов). Тесты 1 – 2. Тесты из условия.
- Группа 1 (100 баллов). Тесты 3 – 42. Каждый тест по отдельности оценивается в 1 балл.

Задача I. Игра в «Кучи»

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Петя готовится к сдаче ЕГЭ по информатике. Он уже отлично умеет решать всю первую часть и сейчас прорабатывает решение второй части заданий.

Двадцать шестая задача посвящена проверке навыков решения задач на поиск выигрышной стратегии в играх с камнями и кучами.

Пете уже наскучили однотипные задачи, и он придумал свою.

Есть n кучек с камнями, в i -ой кучке лежит a_i камней. За одно действие можно добавить один камень в i -ую кучку и одновременно убрать по одному камню из остальных кучек.

Петя хочет проверить, можно ли используя данный алгоритм сделать так, чтобы для любого i в i -ой кучке лежало b_i камней. Ну и главный вопрос – какое наименьшее количество действий ему для этого потребуется?

Формат входных данных

В первой строке расположено одно число n — количество кучек ($1 < n \leq 10^5$).

Во второй строке расположено n чисел a_1, \dots, a_n ($1 \leq a_i \leq 10^5$).

В третьей строке расположено n чисел b_1, \dots, b_n ($1 \leq b_i \leq 10^5$).

Формат выходных данных

Выведите наименьшее количество действий, которое потребуется выполнить, чтобы получить требуемое распределение камней по кучкам. Если это сделать невозможно, выведите «-1».

Примеры

Ввод	Вывод
5 3 5 4 2 6 2 2 1 1 5	3
4 3 5 7 6 3 5 7 4	-1

Система оценки

Тесты к этой задаче состоят из нескольких групп:

- Группа 0 (0 баллов). Тесты 1 – 2. Тесты из условия.
- Группа 1 (20 баллов). Тесты 3–22. В тестах этой группы $n < 100$. Каждый тест по отдельности оценивается в 1 балл.
- Группа 2 (20 баллов). Тесты 23–42. В тестах этой группы $n < 10^4$. Каждый тест по отдельности оценивается в 1 балл.
- Группа 3 (60 баллов). Тесты 43 – 63. В тестах этой группы дополнительные ограничения отсутствуют. Каждый тест по отдельности оценивается в 1 балл.

Задача J. Коробка

Имя входного файла: стандартный ввод

Имя выходного файла: стандартный вывод

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

У девятиклассника Андрея есть младший брат Тимофей, которому недавно исполнилось 2 года. Тимофею родители покупают разные развивающие игрушки, помогающие в познании окружающего мира. Одна из них состоит из коробки размера $N \times M \times K$ и набора из L шаров разного диаметра D_i ($1 \leq N, M, K, L, D_i \leq 100$).

В процессе игры нужно выбрать 2 шара и проверить помещаются ли данные шары в коробку.

Андрей заинтересовался: сколько существует разных наборов из 2 шаров, таких что они удовлетворяют правилам игры. Однако уровень его знаний в программировании недостаточен, поэтому он обращается к Вам за помощью написать такую программу, которая сможет по входным данным определить сколько таких пар существует.

Формат входных данных

В первой строке входного файла расположены числа N , M , и K – размеры коробки. Во второй строке расположено число L – количество шаров. В третьей строке через пробел расположены числа D_1, D_2, \dots, D_L – диаметры шаров. Все числа целые больше 0 и не превосходят 100.

Формат выходных данных

В единственную строку выходного файла необходимо вывести ответ на задачу – количество разных пар шаров, которые можно поместить в коробку.

Примеры

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

Задача К. Важность Марсианских имен

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Марсиане очень гордятся своими именами. Поэтому при знакомстве они обязательно выясняют, чье имя важнее. К сожалению, этот процесс занимает слишком много времени, поэтому марсиане пожелали его автоматизировать.

Марсианские имена представляют собой набор строчных латинских букв, записанных по кругу. Имя следует читать по часовой стрелке, начиная с любой буквы.

Таким образом, количество прочтений марсианского имени совпадает с количеством букв в имени. При сравнении двух имен, марсиане попарно сравнивают всевозможные прочтения первого имени со всевозможными прочтениями второго имени. Прочтение A первого имени является более важным, чем прочтение B второго имени, если A лексикографически больше B . Для каждого сравнения марсиане запоминают, прочтение какого имени оказалось важнее. Более важным считается имя, прочтения которого чаще оказывались важнее.

Вам требуется определить итоговый результат сравнения имен, то есть количество раз, которое прочтение первого имени оказывалось важнее прочтения второго имени, и

количество раз, которое прочтение второго имени оказывалось важнее прочтения первого имени.

Формат входных данных

На входе даны две строки, длиной от 2 до 10^4 символов.

В i -ой строке записано одно из прочтений имени i -ого марсианина. Гарантируется, что имена марсиан не совпадают, то есть никакое прочтение имени первого марсианина не совпадает ни с каким прочтением имени второго марсианина.

Все буквы в именах марсиан – латинские строчные буквы.

Формат выходных данных

В первой строке выведите, сколько раз в результате сравнения имен прочтение первого имени оказывалось важнее прочтения второго имени.

Во второй строке выведите, сколько раз прочтение второго имени оказывалось важнее прочтения первого имени.

Примеры

<i>Ввод</i>	<i>Вывод</i>
ulk ku	4 2

Пояснение к примеру

Первое имя представляет собой буквы «u», «l» и «k», записанные по кругу по часовой стрелке.

Возможные прочтения этого имени: «ulk», «lku» и «kul».

Возможные прочтения второго имени: «ku», «uk».

Сравнения:

1. «ulk» > «ku»
2. «ulk» > «uk»
3. «lku» > «ku»
4. «lku» < «uk»
5. «kul» > «ku»
6. «kul» < «uk»

Прочтения первого имени оказывались важнее 4 раза, прочтения второго – 2 раза.

Задача L. Защита детей

Имя входного файла: стандартный ввод

Имя выходного файла: стандартный вывод

Ограничение по времени: 1 секунда

Ограничение по памяти: 256 мегабайт

Для защиты детей от вредоносной информации в некоторой стране вводится система защиты детей от нежелательной информации.

Как только система была введена в строй и дети стали счастливыми от того, что теперь у них нет доступа к нежелательной информации, коварные нарушители придумали способ борьбы с системой – они стали некоторые буквы латинского алфавита заменять на цифры.

Все комбинации были раскрыты, замены осуществляются следующим образом: $e \Rightarrow 3$, $o \Rightarrow 0$, $i \Rightarrow 1$, $t \Rightarrow 7$, $a \Rightarrow 4$, $s \Rightarrow 5$.

Однако быстро исправить систему невозможно. Но как здорово, что в Школе программиста учитеесь именно Вы.

Напишите программу, которая определяет, есть ли в содержимом текста запрещенное слово?

Формат входных данных

В первой строке расположена строка, содержащая некоторый текст сайта, подлежащего проверке.

Во второй строке расположена строка – запрещенное слово.

Гарантируется, что длина каждой строки не превышает 100.

Формат выходных данных

Выведите «YES», если запрещенное слово встречается в тексте с проверяемого сайта и «NO» в противном случае.

Примеры

<i>Ввод</i>	<i>Вывод</i>
inah0leinthegroundthereliv3d ah0bb1t hobbit	YES
whath4v3igotinmypocket handses	NO
whath4veig0t1nmy0ck37 knife	NO
wh4thav31go71nmy0ck3t stringofnothing	NO