

А. Простой номер

Ограничение времени: 2с

Ограничение памяти: 256МБ

Сотовый оператор планирует ввести новую услугу для любителей математики - номера телефонов, которые представляются в виде простых чисел! Для помощи в реализации задуманного организация обратилась в Кружок Информатики и Робототехники. Вам, как самым опытным участникам Кружка, поручили эту задачу.

Входные данные

Строка, состоящая из цифр, длиной от 1 до 11 символов - номер телефона.

Выходные данные

Если номер телефона можно разделить на последовательность простых чисел, выведите «YES» (без кавычек), иначе выведите «NO» (также без кавычек).

Пример

Входные данные	Выходные данные
311719	YES
424242	NO

В. Самый популярный язык программирования

Ограничение времени: 2с

Ограничение памяти: 256МБ

В Кружке Информатики и Робототехники любят разные языки программирования. Однажды Руководителю Кружка стало интересно, какой язык участники Кружка любят больше всего. Он провел опрос, в котором каждый участник написал название своего самого любимого языка.

Вам, как самым опытным участникам Кружка, доверили подвести итоги опроса.

Входные данные

В первой строке дано целое число N ($1 \leq N \leq 1000$) - количество участников опроса. Далее в N строках приведены названия языков программирования - строки длиной от 1 до 100 символов, содержащие заглавные и строчные латинские буквы, цифры, пробелы и некоторые знаки из списка `+ - = # * /`

Гарантируется, что название языка не начинается и не заканчивается пробелом.

Выходные данные

Выведите название языка программирования, который любит наибольшее количество участников. Если таких языков несколько, то выведите «No winner» (без кавычек).

При подведении итогов регистр символов, наличие и количество пробелов имеет значение, поэтому «JavaScript», «Java Script» и «javascript» считаются разными языками.

Пример

Входные данные	Выходные данные
10 Java java python c++ java script python c# python pascal java	python

С. Коробка в коробке

Ограничение времени: 2с

Ограничение памяти: 256МБ

В Клубок Информатики и Робототехники купили новое оборудование! Оно очень дорогое и хрупкое, поэтому для транспортировки его нужно как следует упаковать.

Было решено упаковать оборудование в несколько вложенных друг в друга коробок. Вкладываемая коробка должна полностью помещаться во внешнюю (то есть ее длина ширина и высота должны быть строго меньше соответствующих размеров внешней коробки). Коробки можно поворачивать, но при этом стороны вкладываемых коробок обязательно должны быть параллельны друг другу - так меньше вероятность, что упаковка помнется. Для большей надежности нужно взять как можно больше коробок.

Для упаковки оборудования подойдет не каждая коробка, поэтому участники Клубка составили список, какие подходящие коробки сейчас есть в наличии. А теперь вам, как самым опытным участникам, нужно из них выбрать подходящие.

Входные данные

В первой строке дано целое число N - количество коробок ($1 \leq N \leq 100$). Далее в N строках даны размеры коробок - целые числа H_i, W_i, D_i ($1 \leq H_i, W_i, D_i \leq 100$).

Выходные данные

В первой строке выведите число M - количество коробок, которые нужно взять. Далее в M строках выведите номера коробок (нумерация начинается с 1).

Коробки выводите в порядке вкладывания - от меньшей к большей.

Пример

Входные данные	Выходные данные
10	3
3 1 2	1
5 2 8	2
1 1 4	7
3 2 2	
6 8 2	
3 1 3	
9 9 9	
8 7 2	
5 5 4	
4 2 7	

D. Кубики

Ограничение времени: 2с

Ограничение памяти: 256МБ

В Кружке Информатики и Робототехники собрали робота для окраски простых деревянных кубиков. Робот берет один кубик, окрашивает все его грани и кладет окрашенный кубик на место.

Определите, за какое время робот окрасит данное количество кубиков.

Входные данные

В строке даны целые числа, разделенные одним пробелом - N , M и K ($1 \leq N, M, K \leq 1000$) - количество кубиков, время (в секундах), требуемое на окрашивание одной грани и время (также в секундах), требуемое для того, чтобы взять или положить кубик.

Перед началом работы робот не держит кубик.

Выходные данные

Выведите через пробел три целых числа - количество часов, минут и секунд, требуемое на окрашивание всех кубиков.

Работа считается законченной, когда последний окрашенный кубик будет положен на место.

Пример

Входные данные	Выходные данные
50 12 5	1 8 20

Е. Степень палиндрома

Ограничение времени: 2с

Ограничение памяти: 256МБ

В Кружке Информатики и Робототехники изучают в том числе строки и их свойства. Недавно Руководитель Кружка рассказал ребятам о палиндромах - строках, которые читаются одинаково слева направо и справа налево. Чтобы изучать их было интереснее, Руководитель придумал свойство *степень палиндрома*.

У обычного палиндрома *степень* 1. Если его «половины» тоже являются палиндромами, то *степень* равняется 2. Если половины половин тоже палиндромы, то 3 и так далее. Если строка не является палиндромом, то ее *степень* равняется 0.

Чтобы помочь начинающим, Руководитель обратился к вам, самым опытным участникам кружка, с просьбой помочь с вычислением *степеней* палиндромов.

Входные данные

Дана строка, состоящая из строчных латинских букв, длиной от 1 до 10^6 символов.

Выходные данные

Выведите *степень* палиндрома данной строки.

Пример

Входные данные	Выходные данные
abacaba	3

Г. Растеризация

Ограничение времени: 2с

Ограничение памяти: 256МБ

Для оформления программ в Кружке Информатики и Робототехники хотят использовать собственный шрифт. Но чтобы не хранить растровые изображения всех символов (ребята знают, что память нужно экономить), они решили использовать векторный шрифт собственного формата. Каждый символ в шрифте задается набором отрезков.

Шрифт ребята уже разработали, теперь вам, как самым опытным участникам Кружка, предстоит реализовать вывод текста с помощью символов этого шрифта.

Входные данные

В первой строке дано целое число N ($1 \leq N \leq 26$) - количество символов в шрифте.

Далее для каждого символа шрифта в отдельной строке через пробел даны строчная латинская буква C_i , которую представляет символ, и количество отрезков M_i ($1 \leq M_i \leq 10$).

Гарантируется, что все символы C_i различны.

Далее в M_i строках даны описания отрезков, составляющих символ - через пробел даны буква S_j («H» для горизонтального отрезка и «V» для вертикального), целые числа x_j , y_j , d_j ($0 \leq x_j, y_j \leq 50$, $1 \leq d_j \leq 50$) - отступы по горизонтали и вертикали начала отрезка и его длина. Точка (0,0) находится в верхнем левом углу изображения символа.

Для горизонтального отрезка конец располагается правее, а для вертикального - ниже начала.

Далее дана строка длиной от 1 до 20 символов, которую нужно вывести. Гарантируется, что строка состоит только из символов, которые есть в шрифте.

Выходные данные

Выведите написание данной строки данными символами. Для точек, занятых символом, выведите «*» (звездочку), для пустых точек выведите «.» (точку). Вокруг текста не должно быть пустых строк и столбцов (то есть полностью состоящих из точек).

Символы выводятся слева направо и начало следующего символа не может быть левее начала предыдущего. Символы нужно расположить максимально близко друг к другу, то есть следующий символ должен занимать максимально левое положение относительно предыдущего, но без взаимных касаний (по вертикали, горизонтали и диагонали). Символы вертикально выравниваются по нижнему краю изображения.

Кубок Псковской области по программированию среди школьников — 2020
Финальный раунд, 06 ноября 2020 года

Пример

Входные данные	Выходные данные
<pre>4 f 3 H 0 0 18 V 0 0 8 H 0 3 3 o 4 H 1 0 2 V 0 1 3 H 1 4 2 V 3 1 3 n 3 H 0 0 3 V 0 0 5 V 3 1 4 t 2 H 0 0 5 V 2 0 6 font</pre>	<pre>***** * ****** ***..**..**..*.. *...*..*..*..*.. *...*..*..*..*.. *...*..*..*..*.. *...**..*..*..*.. *.....**..*..*..*.. *****</pre>
<pre>4 f 4 H 0 0 8 H 0 2 2 V 0 0 6 V 7 0 3 o 4 H 0 0 3 H 0 3 3 V 0 0 4 V 2 0 4 n 3 H 0 0 7 V 0 0 6 V 6 0 6 t 2 V 1 0 4 H 0 0 3 font</pre>	<pre>*****.***** **.* **.***.*.***.* *...*..*..*..*..* *...*..*..*..*..* *...***..*..*..* *.....**..*..*..* *****</pre>

Г. Непростой путь

Ограничение времени: 2с

Ограничение памяти: 256МБ

Кружок Информатики и Робототехники готовится к соревнованию по робототехнике. На соревновании роботы должны пройти по местности с препятствиями. Карта местности представляет собой прямоугольник, разделенный на квадратные клетки, некоторые клетки свободны, некоторые заняты препятствием, то есть непроходимы. Робот должен следовать кратчайшим маршрутом, не проходить по одной клетке дважды, а также не выходить за пределы карты. Карта местности известна, но на ней отмечена только точка начала маршрута, конечную точку объявят только на самом соревновании.

Для участия в соревнованиях участники Кружка подготовили шагающего робота, но у него есть неисправность - каждый раз, когда он проходит от начала движения количество шагов, равное простому числу, есть вероятность потери связи. Ребята не успевают найти и устранить причину неисправности до начала соревнований, поэтому вам, как самым опытным участникам, поручили составить карту для оператора робота, на которой отметить участки, где с роботом может быть потеряна связь.

Робот может ходить по карте местности только по вертикали и горизонтали.

Входные данные

В первой строке через пробел даны целые числа N и M ($1 \leq N, M \leq 100$) - высота и ширина карты местности. Далее в N строках по M символов дана карта, каждый символ обозначает:

- Пустая клетка - «.»
- Клетка с препятствием - «#»
- Точка начала маршрута - «S»

Выходные данные

Выведите карту местности из входных данных, на которой отмечены символом «*» клетки, где возможна потеря связи с роботом.

Пример

Входные данные	Выходные данные
5 7#.# ..#.#.# .S#...# ###.#.##.#	***.#.# *.#*#.# .S#.*.# ###*#.# *...#.#

Н. Равные треугольники

Ограничение времени: 2с

Ограничение памяти: 256МБ

В Кружке Информатики и Робототехники продолжают разрабатывать графический редактор, который может распознавать, что рисует пользователь. Чтобы сэкономить память, ребята хотят не хранить дважды одинаковые фигуры. Для начала решено было подготовить реализацию для треугольников.

Входные данные

В первой строке через пробел даны целые числа a, b, c ($1 \leq a, b, c \leq 1000$) - длины сторон первого треугольника.

Во второй строке через пробел даны целые числа n, m, k ($1 \leq n, m, k \leq 1000$) - длины сторон второго треугольника.

Гарантируется, что треугольники с данными длинами сторон существуют.

Выходные данные

Если данные треугольники равны, выведите «YES» (без кавычек), иначе выведите «NO» (также без кавычек).

Пример

Входные данные	Выходные данные
1 2 3 2 3 1	YES
3 5 2 8 6 4	NO

I. Поворот

Ограничение времени: 2с

Ограничение памяти: 256МБ

В кружке информатики и Робототехники изучают в том числе и алгоритмы шифрования. Сейчас они рассматривают такой алгоритм.

У отправителя и получателя есть одинаковые картонные диски, на которых по кругу нанесены все 26 букв латинского алфавита (цифры и знаки препинания при передаче информации не используются, а пробелы пропускаются). Для того, чтобы зашифровать сообщение, нужно установить начальное положение - это положение буквы **a**. Далее для каждого символа шифруемого текста нужно поворачивать диск по часовой стрелке, записывая для каждого символа, на сколько букв повернулся диск относительно предыдущего положения. Последовательность поворотов и будет шифром.

Входные данные

В первой строке дана последовательность из 26 строчных латинских букв - символы, нанесенные на диск, в порядке обхода по часовой стрелке.

Во второй строке дана последовательность символов, состоящая из строчных латинских букв и пробелов длиной от 1 до 1000 символов, которую нужно зашифровать.

Выходные данные

Выведите через пробел последовательность целых чисел - шифр.

Пример

Входные данные	Выходные данные
mbjyalzvepsfonxqgduritwkch hello world	21 9 23 0 7 10 16 7 12 12