



A. Hello, world

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Вы знаете, что нужно делать.

Входные данные

Нет.

Выходные данные

Выведите «Hello, world!» без кавычек, но с учётом регистра символов и знаков препинания.

Пример

Входные данные	Результат
	Hello, world!



B. Hello, user

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Вы наверняка писали такую программу. Пользователь вводит своё имя, например, «Vasya», а программа отвечает ему «Hello, Vasya!».

Входные данные

В единственной строке указано имя, состоящее из маленьких и больших английских букв и пробелов. Длина имени не меньше 1 символа и не больше 100 символов.

Выходные данные

В единственной строке выведите фразу-приветствие.

Пример

Входные данные	Результат
Vasya Pupkin	Hello, Vasya Pupkin!
Testoviy Test Testovich	Hello, Testoviy Test Testovich!
world	Hello, world!

Подсказка

Тем, кто участвует в подобных соревнованиях впервые, следует обратить внимание на то, что программа должна жестко следовать указанной спецификации формата входных и выходных данных.

В частности, нельзя выводить ничего, кроме того, что сказано вывести в спецификации выходных данных. В этой задаче, например, *нельзя* выводить вопрос к пользователю (например, «Input your name:»). Ваше решение проверяется автоматической проверяющей системой, и общаться с ней как с человеком не нужно.

С другой стороны, вам гарантируется, что данные, поступающие на вход, тоже строго соответствуют спецификации. Следовательно, проверять их на корректность, или, тем более, просить пользователя повторить ввод, если они не корректны, не следует. Помните — нет никакого пользователя.

Входные данные следует читать как с клавиатуры (`scanf`, `cin`, `System.in`), а выходные — выводить как на экран (`printf`, `cout`, `System.out`). Говоря более формально, следует работать со стандартными потоками ввода и вывода.



С. Смешная функция

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Даны два целых числа, A и B . Выведите значение функции:

$$f(A, B) = A - B - A + B + A - A + B + A + B + B - A + B - B - B - B + A$$

Входные данные

В единственной строке находятся два целых числа A и B ($-10^9 \leq A, B \leq 10^9$), разделенные пробелом.

Выходные данные

Выведите единственное целое число — значение функции.

Пример

Входные данные	Результат
17 25	42



D. Диплом

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Помните Васю Пупкина? Ещё недавно он только поступил в X-OSPU (eXtreme Open Source Programming University), а вот прямо сейчас он уже несёт свой диплом на переплёт. Но совсем не о дипломе думает Вася. Только что на его смартфон пришло уведомление о том, что доступно для загрузки долгожданное обновление до RobotOS 9.0 Pumperkin. «Не буду ждать возвращения домой, поставлю прямо сейчас», — решил Вася.

И вот, Вася идёт, неся в одной руке пачку только что распечатанных страниц диплома, а другой рукой запускает процесс обновления. И в этот момент Вася в 14 835-ый раз в своей жизни вспомнил, почему он уже 14 834 раза обещал себе больше никогда не залипать в телефон на ходу. Вася бежал наперегонки с ветром, ловил страницы руками, придавливал ногами, сбивал с ног идущих по дороге студентов, едва разминулся с деканом...

Когда погоня наконец была закончена, и Васе удалось собрать все страницы, перед ним оказалась стопка из N страниц диплома, пронумерованных в строгом порядке от 1 до N , но разложенных в строгом беспорядке. Помогите Васе отсортировать страницы в правильном порядке, чтобы он мог поскорее начать установку долгожданного обновления для своего смартфона.

Входные данные

В первой строке дано число N ($1 \leq N \leq 10^3$) — количество страниц в дипломе Васи. Во второй строке дано N чисел, разделённых пробелами — номера страниц, в том порядке, в котором они оказались после окончания погони.

Выходные данные

В единственной строке выведите N чисел, разделённых пробелами — номера страниц в правильно отсортированном порядке.

Пример

Входные данные	Результат
5 3 2 4 5 1	1 2 3 4 5



Е. О девочках

Ограничения: время работы — 5 секунд, использование памяти — 256МБ

Саша и Таня — девочки. А девочки, как вы все знаете, любят играть в игры. И даже сейчас, надо же, пришла им в голову идея сыграть в игру.

У девочек есть набор из N целых положительных чисел. Так как набор этот был найден в интернете (а наборам чисел из Интернета верить нельзя ни в коем случае!), он странный — в нём точно найдется как минимум одна пара чисел отличающихся в точности на единицу. Если Таня правильно назовёт наибольший общий делитель (GCD) всех этих чисел, то победит Саша, а если не назовет или назовет неправильно — победит Таня.

Таня очень любит свою подругу, поэтому хочет, чтобы она победила. Помогите ей это сделать — найти наибольший общий делитель данного набора.

Входные данные

В первой строке входных данных указано число N ($2 \leq N \leq 10^6$) — количество чисел в наборе. В следующей строке указаны числа из набора, разделенные пробелом. Каждое число в наборе находится в промежутке $[1; 10^9]$. О наборе точно известно, что в нём найдется как минимум одна пара чисел, разница между которыми составляет в точности 1.

Выходные данные

Одно число — ответ на вопрос задачи.

Пример

Входные данные	Результат
2 1 2	1



Ф. До конца пары

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Вася Пупкин уже третий год учится в X-OSPU (eXtreme Open Source Programming University). Ещё в прошлом году Вася перестал ходить на первые пары, а теперь он уж совсем редко показывается в университете. Настолько редко, что однажды, сидя на скучной лекции (на которую просто необходимо было прийти, если хочешь получить зачёт), он понял, что уже не помнит, когда заканчиваются пары.

Конечно, ходить на занятия чаще и постепенно снова запомнить — это не вариант. Вместо этого Вася решил доработать свою программу с расписанием занятий для RobotOS, которую он написал ещё на первом курсе. Просто показывать время конца пары и заставлять пользователя самостоятельно производить сложные вычисления в уме — это не хорошо, решил Вася, поэтому его программа должна сразу выводить, сколько времени осталось терпеть до конца пары. К сожалению, у Васи сейчас полный завал с долгами, а ещё и нужно обновлять свой смартфон до RobotOS 7.1 Nutella, так что он просит вас помочь с новым функционалом.

На вход вашего модуля подаётся текущее время и время окончания пары. Модуль должен вернуть количество времени, оставшегося до конца пары. Ночью занятия в X-OSPU не ведутся, поэтому переходов через полночь точно не будет.

Входные данные

В первой строке указано текущее время, а во второй — время окончания пары. И то, и другое время дано в формате ЧЧ:ММ, то есть часы и минуты с ведущими нулями, разделённые двоеточием. Часы, как вы можете знать, могут находиться в диапазоне от 0 до 23 включительно, а минуты — от 0 до 59 тоже включительно. Гарантируется, что текущее время не больше, чем время окончания пары.

Выходные данные

Выведите количество времени, оставшегося до конца пары в формате ЧЧ:ММ, то есть количество часов и минут с ведущими нулями, разделённые двоеточием.

Пример

Входные данные	Результат
08:42 09:50	01:08
12:37 13:15	00:38



G. Мобильный интернет

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

В этом году Вася Пупкин, студент X-OSPU (eXtreme Open Source Programming University), уже четверокурсник. А что это значит? Правильно, это значит, что он стал ещё меньше ходить на пары. Да, так мало, что с некоторыми преподавателями он знакомится прямо на... Нет, что вы, не на экзамене. На модульном контроле.

Так как же Васе написать контрольную по предмету «Экономика Разработки Усовершенствований, Направленных На Дополнительную Автоматизацию» («ЭРУННДА»), если он не успел выучить даже название предмета? Мобильный интернет приходит на помощь! Вася, пряча смартфон под партой, уже успел нагуглить два подходящих конспекта. Но есть проблема: конспекты, похоже, сделаны картинками, поэтому файлы — большие. А с тех пор, как Вася скачал обновление для своего смартфона до RobotOS 8.1 Octopus через своего оператора deathcell:(, у него каждый мебибайтик на счету. «Скачаю тот, что меньше», — решил Вася.

Но сравнить размеры файлов оказалось не так-то просто. На одном сайте размер указан в мегабайтах, а на другом — в мебибайтах. При этом сами числа близкие, так что без точных подсчётов не обойтись. К сожалению, после обновления калькулятор у Васи на смартфоне почему-то всё время падает, поэтому Вася просит вас помочь с расчётами.

Вам даны размеры обоих файлов. Определите, какой из них меньше, либо что они равны.

Входные данные

В первой строке указан размер первого файла, а во второй — размер второго файла. Оба размера заданы в формате «целое число, пробел, единица измерения». Единица измерения может быть B, KB, KiB, MB, MiB, GB, GiB, TB, TiB, PB, PiB, EB или EiB (смотрите таблицу на следующей странице). Размеры каждого из файлов строго меньше 8 эксбибайт.

Выходные данные

Выведите один символ — знак отношения, который нужно поставить между размером первого файла и размером второго файла: <, > или =.

Примеры

Входные данные	Результат
42 MB 41 MiB	<
1000 KiB 1024 KB	=
41 MiB 42991616 B	=



Справочная информация

Приставка	Множитель	
K (kilo, кило)	10^3	1 000
Ki (kibi, киби)	2^{10}	1 024
M (mega, мега)	10^6	1 000 000
Mi (mebi, меби)	2^{20}	1 048 576
G (giga, гига)	10^9	1 000 000 000
Gi (gibi, гиби)	2^{30}	1 073 741 824
T (tera, тера)	10^{12}	1 000 000 000 000
Ti (tebi, теби)	2^{40}	1 099 511 627 776
P (peta, пета)	10^{15}	1 000 000 000 000 000
Pi (pebi, пеби)	2^{50}	1 125 899 906 842 624
E (exa, экса)	10^{18}	1 000 000 000 000 000 000
Ei (exbi, эксби)	2^{60}	1 152 921 504 606 846 976



Н. Первые пары

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

В прошлом учебном году Вася Пупкин начал обучение в X-OSPU (eXtreme Open Source Programming University), а в этом году он уже второкурсник. И теперь он, как и многие другие студенты, почувствовавшие себя старожилками, ленится ходить на первые пары. Но есть две проблемы. Во-первых, в X-OSPU очень строго относятся к срокам сдачи лабораторных и снижают баллы, если не сдать их вовремя. Поэтому всё-таки придётся ходить на все лабораторные. А во-вторых, Васе очень нравится преподаватель по имени Черя, поэтому на все его занятия он тоже хочет ходить.

В прошлом году Вася научился программировать под свой смартфон на RobotOS, поэтому теперь он хочет написать программу, которая будет автоматически устанавливать будильники на его смартфоне в зависимости от расписания (чтобы автоматизировать прогуливание, так сказать). А для этого необходимо сначала определить, в какие дни нужно приходить на первую пару, а в какие — нет.

Помогите Васе написать этот модуль для его программы. На вход модуля поступает расписание занятий на неделю, а на выход он должен выдать список дней недели, в которые таки нужно идти на первую пару.

Входные данные

На вход программы поступает расписание занятий. Оно состоит из пяти блоков — по одному для каждого рабочего дня недели. Блоки разделяются пустой строкой, но после последнего блока пустой строки нет.

Блок начинается с отдельной строки-заголовка, в которой указан день недели («Monday», «Tuesday», «Wednesday», «Thursday» или «Friday»). Блоки всегда упорядочены от понедельника к пятнице. За строкой-заголовком следует одна или более строк, описывающих занятия в этот день.

Каждое занятие описывается одной строкой, которая содержит через пробел: номер пары (целое число от 1 до 9), тип занятия («lec» — лекция, «prc» — практика или «lab» — лабораторная) и имя преподавателя (от 1 до 20 маленьких и больших английских букв). Занятия упорядочены по номерам пар, но некоторые пары могут быть пропущены («окна»).

Выходные данные

Выведите список дней недели, в которые нужно идти на первые пары, упорядоченный от понедельника к пятнице. Названия дней недели должны быть такими же, как во входных данных, и их следует выводить по одному в строке.

Если ни в один день не нужно идти на первую пару, выведите единственную строку «Truant!» (что значит «Прогульщик!»).

Пример

Входные данные	Результат
Monday 1 prc Lyubchenko 2 prc Черя 3 lec Ruvinskaya	Wednesday Thursday Friday
Tuesday 2 lab Ruvinskaya 3 lec Krisilov	
Wednesday	

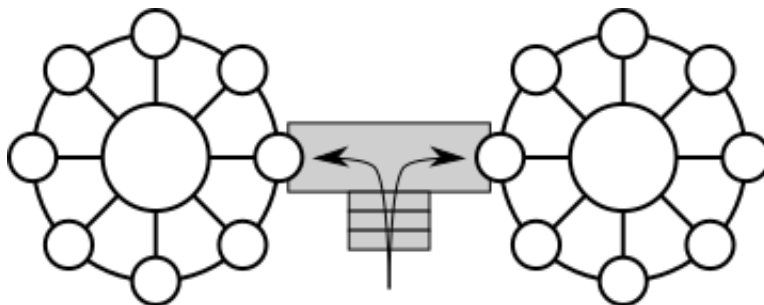


Входные данные	Результат
<p>1 lab Kavickaya 2 lec Zinovatnaya 3 lec Lyubchenko</p> <p>Thursday 1 lec Chepa 2 lab Chepa 3 lab Zinovatnaya</p> <p>Friday 1 lab Lyubchenko 2 prc Ruvinskaya 3 prc Zinovatnaya</p>	
<p>Monday 1 lec Onishenko 2 prc Komlevaya 3 lab Kungurtsev</p> <p>Tuesday 1 lec Kungurtsev 2 lab Onishenko 3 lec Chepa</p> <p>Wednesday 2 lab Chepa 3 prc Kungurtsev</p> <p>Thursday 1 prc Onishenko 2 prc Chepa 3 lec Komlevaya</p> <p>Friday 1 lec Sinegub 2 lab Komlevaya 3 prc Sinegub</p>	<p>Truant!</p>

I. Карусель

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Два друга решили покататься на каруселях. В парке были две карусели, соединенные общей посадочной площадкой, и мальчики решили кататься одновременно на разных.



Время их катания ограничено 1 часом. Заданы скорости вращения каруселей V_1 и V_2 в виде целого числа оборотов в час. Известно, что мальчики начали катание одновременно. Определить, через какое минимальное время после начала движения они снова окажутся на минимальном расстоянии друг от друга.

Входные данные

В единственной строке через пробел записаны два целых положительных числа V_1 и V_2 ($1 \leq V_1, V_2 \leq 30$).

Выходные данные

Ответ на вопрос задачи в виде несократимой обыкновенной дроби P/Q без незначащих нулей в числителе и знаменателе.

Пример

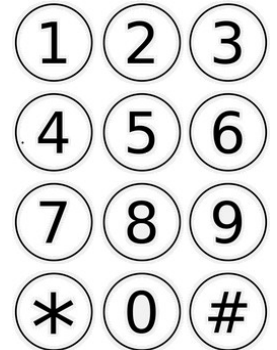
Входные данные	Результат
2 3	1/1
6 8	1/2
1 1	1/1
30 30	1/30

Ж. Телефонные номера

Ограничения: время работы — 1 секунда, использование памяти — 256МБ



Никита со своей девушкой гуляли по парку и случайно нашли на одной из лавочек у пруда мобильный телефон. Без долгих раздумий ребята решили вернуть телефон хозяину. Для этого они задумали позвонить на один из номеров в телефонной книжке и сообщить о том, что телефон у них. Но вот незадача – в телефоне огромное количество номеров. И, почему-то, ни одного имени – просто номера.



На какой же номер звонить? На случайный, сказали бы вы. Но ребята не любят случайные числа и хотят, чтобы их выбор был на чём-то основан. Выбор пал на самый оптимальный для набора номер. Номер

должен быть таким, чтобы как можно меньше водить пальцами при наборе (движение пальца до самой первой кнопки не учитывается). Расстояние между кнопками равно $|x_1 - x_2| + |y_1 - y_2|$. Кнопки соответствуют стандартному мобильному циферблату, как на картинке. Например, кнопка 1 имеет координаты $(0, 0)$, а кнопка 0 – $(3, 1)$.

Если оптимальных номеров несколько, ребята выберут тот, который лексикографически меньший.

Помогите ребятам определить на какой номер звонить, так как у них мало времени – они еще хотят успеть погулять и покушать мороженое.

Входные данные

Первая строка содержит числа N ($1 \leq N \leq 10^4$) — количество номеров, и M ($1 \leq M \leq 20$) — количество цифр в номерах. Далее идут N строк длиной по M символов — номера телефонов. Все символы являются цифрами.

Выходные данные

Выведите номер, на который необходимо позвонить.

Пример

Входные данные	Результат
3 3 093 140 204	093
2 4 9241 1138	1138



К. Упертость Марксинуса

Ограничения: время работы — 1 секунда, использование памяти — 256МБ

Марксинус днями и ночами не спит, только и делает, что думает о ней... О единице в числе. Да, 1 — наилучшая цифра, по мнению Марксинуса, и он уверен, что такая прекрасная цифра должна встречаться как можно больше раз в числах.

Марксинус довольно упертый и доверяет только вычислительной технике, с людьми дело иметь он не особо хочет, особенно если тема спора затрагивает его любимую... Да-да, конечно же единицу. А упертость Марксинуса заключается в том, что он уверен, что чисел от 1 до 10^n , в записи которых встречается единица, больше, чем тех, в которых она не встречается.

Докажите или опровергните его мысль. В случае, если Марксинус прав, выведите «YES», иначе — «NO».

Входные данные

Одно целое число n ($1 \leq n \leq 10^6$).

Выходные данные

В случае, если Марксинус прав, выведите «YES», иначе — «NO».

Пример

	Входные данные	Результат
1		NO

L. Бинарный поиск

Ограничения: время работы — 5 секунд, использование памяти — 256МБ

Сегодня у студентов группы PA-101 университета X-OSPU лекция по Теории алгоритмов. Их преподаватель, Николай Алгоритмович, рассказывает им про бинарный поиск. Вообще говоря, бинарный поиск — довольно абстрактный алгоритм, опытные олимпиадники даже умеют применять технику, называемую «бинарный поиск по ответу»... Но, начнём с простого. В частности, бинарный поиск позволяет довольно быстро (за $O(\log_2(n))$) проверить, есть ли в отсортированном массиве заданное число.

Николай Алгоритмович привёл ребятам вот такой пример реализации бинарного поиска:

```
bool check(const int* a, int n, int x) {
    int l = 0, r = n - 1;
    while (l <= r) {
        int m = (l + r) / 2;
        if (a[m] == x) return true;
        if (a[m] < x) l = m + 1;
        else r = m - 1;
    }
    return false;
}
```

Разумеется, сначала массив нужно отсортировать, но студенты PA-101 уже знают, что в библиотеке `<algorithm>` их любимого языка C++ уже есть готовая функция сортировки, так что отсортировать массив можно одной строчкой кода:

```
std::sort(a, a + n);
```

Для того, чтобы проверить, как студенты усвоили материал, Николай Алгоритмович задал им написать дома следующую программу. Программа получает на вход массив A , состоящий из N чисел (первое число имеет индекс 1, последнее — N), и Q запросов, каждый из которых состоит из одного числа X_i . Программа должна ответить на каждый запрос. Если в заданном массиве есть число с индексом X_i , то программа должна вывести индекс этого числа в массиве A , иначе — вывести «-1».

Помогите студентам справиться с домашним заданием. Разумеется, вы можете воспользоваться их конспектом.

Входные данные

В первой строке находится число N ($1 \leq N \leq 10^5$) — количество элементов в массиве A . Во второй строке находится N целых чисел a_i ($-10^9 \leq a_i \leq 10^9$) — элементы массива. В третьей строке находится число Q ($1 \leq Q \leq 10^5$) — количество запросов. В четвёртой строке находится Q целых чисел X_i ($-10^9 \leq X_i \leq 10^9$) — сами запросы.

Выходные данные

Для каждого из Q запросов выведите в отдельной строке одно число — ответ на соответствующий запрос.



Пример

Входные данные	Результат
3	3
1 2 3	1
5	2
3 1 2 5 42	-1
	-1