Об олимпиаде

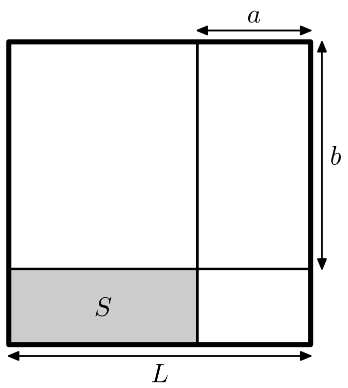
В этой интернет-олимпиаде мы тестируем сервис отправки сообщений для ИОИП.
На первые пять попыток, принятых на проверку, по каждой задаче вы получите свои баллы.
Вы можете решать все пять задач.

Задача A. Разрезание заготовки

Имя входного файла:	`cut.in`
Имя выходного файла:	`cut.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Юной помощнице великого художника Оле надоело все время быть в тени своего мастера, и она решила написать свой шедевр, воспользовавшись тем, что художник решил днем вздремнуть. Для этого она взяла квадратную заготовку, которыми была полна мастерская. Однако заготовка показалась ей слишком велика, да и не по статусу ей было расписывать настолько большое полотно. Поэтому Оля решила отрезать от заготовки полосу шириной a справа, сделав вертикальный разрез.

Теперь полотно показалось ей слишком высоким, и она сделала горизонтальный разрез, убрав с заготовки сверху полосу высотой b. Для того, чтобы убедиться, что полученное полотно подойдет для ее работы, она вычислила его площадь S и записала ее на листок бумаги.



Через несколько часов работы Оля сочла свое творение законченным и достойным внимания публики, поэтому ушла с ним на фестиваль в центр города.

Проснувшись, великий художник сразу все понял — таких случаев в его жизни было много. Для того, чтобы восстановить запасы заготовок, ему теперь требуется купить новую квадратную заготовку, в точности такую же, какую использовала Оля. Единственное, что мастеру удалось узнать об изначальных размерах полотна, это числа a, b, S — они были записаны на найденных в мастерской клочках бумаги.

Ситуация осложняется тем, что в продаже присутствуют заготовки только с длинами сторон, которые выражаются целыми числами. Напишите программу, которая по числам a, b и S определит имеет ли смысл идти в магазин за новой заготовкой и найдет длину стороны заготовки.

Формат входного файла

Единственная строка входного файла содержит три целых числа a, b и S ($0 \le a, b \le 10^4$, $1 \le S \le 10^8$).

Формат выходного файла

Если соответствующая заготовка есть в продаже, то в выходной файл выведите единственное целое число — длину ее стороны L.

Если же такой заготовки при заданных параметрах не может существовать, или ее нет в продаже, выведите в выходной файл число «`-1`».

Примеры

`cut.in`	`cut.out`
`1 1 1`	`2`
`2 3 12`	`6`
`1 1 2`	`-1`

Задача B. Верное выражение

Имя входного файла:	`expression.in`
Имя выходного файла:	`expression.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Многие утверждения на первый взгляд кажутся неверными, например, равенство $2 + 2 = 11$. Но оказывается, что в системе счисления с основанием 3 это равенство превращается в верное: $2_3 + 2_3 = 11_3 = 4_{10}$.

Задано выражение A ? $B = C$, где A, B и C — неотрицательные целые числа, а ? — символ сложения, вычитания или умножения. Требуется вывести список всех оснований систем счисления от 2 до 10, для которых данное равенство является верным. Напоминаем, что в системе счисления с основанием n используются только цифры от 0 до $n - 1$.

Формат входного файла

В единственной строке входного файла содержится выражение A ? $B = C$. Вместо символа ? могут стоять символы $+$, $-$ или $*$. Части выражения отделены друг от друга одним пробелом. Длины записей чисел A, B и C не превышают тысячи. В записях этих чисел используются цифры от 0 до 9. Записи чисел A, B и C не содержат избыточных ведущих нулей.

Формат выходного файла

В первой строке выведите единственное число k — число подходящих оснований. Во второй строке выведите k чисел — основания, перечисленные в возрастающем порядке.

Примеры

expression.in	expression.out
`2 + 2 = 11`	`1` `3`
`2 + 2 = 4`	`6` `5 6 7 8 9 10`
`2 * 2 = 5`	`0`

Решения, работающие для A, B и C, длины которых не превышают 9, будут оцениваться в 60 баллов.

Задача C. Княжества

Имя входного файла:	`kingdoms.in`
Имя выходного файла:	`kingdoms.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

> Лествичное право (от старо-славянского «лествица» - лестница) - обычай великокняжеского наследования в Киевской Руси.

В великом государстве Эн правил мудрый правитель Эм. У мудрого правителя Эм было несколько сыновей. И чтобы каждый из сыновей вырос хорошим правителем, решил Эм каждому выделить по княжеству великому в управление. Но не все княжества равны между собой были, ибо часть была в плодородных землях черноземных, часть в лесах темных, часть в степях диких, да и по размеру княжества были различны. Задумался мудрый Эм, и порешил занумеровать все княжества в порядке главенства – самое главное княжество нарек он Первым, следующее – Вторым, и так далее. И княжеств в Энском государстве было бесчисленное множество, и каждое свой номер имело. Старшему наследнику отдал Эм Первое княжество. И Второму своему сыну отдал он Второе княжество, и так далее.

Но тут родилися у Эм внуки, и снова задумался мудрый правитель. Ибо внукам надлежало тоже быть хорошими правителями, а значит и им надобно было выделить княжества. И порешил Эм, что выделит он всем внукам по княжеству, но не в порядке старшинства, а иначе – сперва лучшие княжества среди сыновей в порядке старшинства распределяются, затем следующие княжества сыновьям старшего брата в правление отдаются, также по старшинству распределенные, потом второго брата сыновьям, по старшинству, и так далее. И обрадовался Эм, ибо система такая позволяла и правнукам, и всем прочим потомкам мужеского пола выделять княжества в правление, и по главенству княжества верно распределены были. И когда родился у Эм еще один сын, то все внуки его и правнуки, и остальные потомки мужеского пола править стали княжеством с на единицу большим номером, а сын новорожденный стал править княжеством с номером на единицу большим чем у второго по младшинству сына, ибо никакой внук не правил княжеством более главным, чем каждый из сыновей, а среди сыновей порядок главенства старшинством задается.

Но тут печаль у Эм приключилась. Умер сын его, и великое княжество без правителя осталось. И призадумался Эм, и решил, что если погиб потомок, то княжество, без правителя оставшееся, правитель следующего княжества княжить будет. Так, все следующие потомки мужекого пола станут княжить земли с на единицу меньшим номером, и все княжества верно по главенству распределены будут.

Получилось так, что в великом государстве в каждый момент времени есть порядок на князях, задаваемый следующим образом: сначала правитель Эм, потом его сыновья в порядке старшинства, потом сыновья старшего сына также в порядке старшинства, потом аналогично сыновья второго по старшинству сына Эма и так далее вплоть до самого младшего.

Таким образом, любой князь из более позднего поколения идет после любого князя из более младшего (даже, если первый старше). Например, любой правнук идет после любого внука.

Летописец государства Эн прилежно вел запись всех событий государства, всех смертей и рождений, в хронологическом порядке. В ваши руки попала эта бесценная летопись. Требуется написать программу для обработки этой летописи, которая будет способна определять, кто правит в k-ом княжестве в определенные моменты времени?

Формат входного файла

В первой строчке содержится единственное число n — количество событий и запросов в государстве великом ($1 \le n \le 2000$). Следующие строки содержат описания событий в государстве великом и запросов о том, кто в каком княжестве княжит по одному событию или запросу в строчке и в порядке происшествия данных событий в следующем формате:

- строка вида «+ Father Son» сообщает о великом событии, что у князя с именем «Father» родился сын, которого он нарек «Son»;
- строка вида «- Name» сообщает о великом горе, что умер князь, нареченный «Name»;
- строка вида «? k» есть суть вопрос — кто в данный момент княжит в k-ом по главенству княжестве ($1 \le k \le 2500$).

Никакие два князя за все время существования государства не имели одинаковых имен и все имена князей в летописи данной записаны только строчными и заглавными латинскими буквами и имеют длину не более 20 символов.

Летопись началась вестись в тот момент, когда в государстве жил один единственный мудрый правитель, именовали которого «M», то есть просто Эм, и правил он в Первом княжестве.

Формат выходного файла

На каждый запрос вида «? k» в выходной файл выведите по одной строке, содержащей имя князя, правившего на момент запроса в k-ом по главенству княжестве, или единственный символ «-», если на момент запроса в k-ом княжестве нет правителя, ибо нет такого количества живых князей.

Все ответы выводите на запросы в том порядке, в котором они задаются во входном файле.

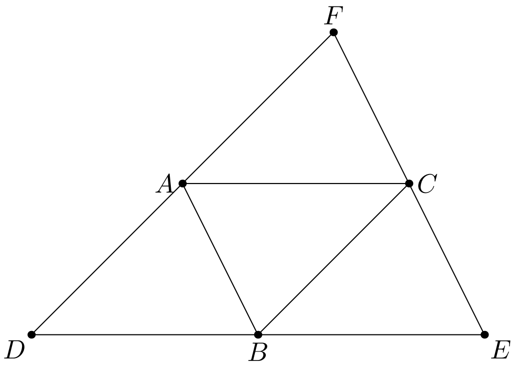
Примеры

kingdoms.in	kingdoms.out
7 + M Ivan ? 1 ? 2 ? 3 - M ? 2 ? 1	M Ivan - - Ivan
10 + M Jaroslav + M Vladimir + Vladimir Dimitry ? 2 ? 3 ? 4 + Jaroslav Svjatoslav ? 4 ? 5 ? 1	Jaroslav Vladimir Dimitry Svjatoslav Dimitry M

Задача D. Треугольники

Имя входного файла:	`triangles.in`
Имя выходного файла:	`triangles.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Недавно Вася объяснил своему младшему брату Пете основы работы с циркулем и линейкой. В качестве примера он показал, как по данному треугольнику построить такой, что середины его сторон совпадают с вершинами первого. Такое действие он назвал «расширением» треугольника.



Например, на рисунке треугольник DEF является расширением треугольника ABC.

Для закрепления материала Вася решил дать брату какое-нибудь простое задание на точность построения геометрических фигур. Немного поразмыслив, он нарисовал на листе бумаги n треугольников и назвал «шагом» однократное расширение всех треугольников. После этого он попросил Петю узнать, через какое число шагов найдется пара пересекающихся треугольников.

Два треугольника пересекаются, если существует точка, лежащая внутри или на границе каждого треугольника. В частности, если один треугольник находится внутри другого, то такие треугольники пересекаются.

Петя слушал умного брата не очень внимательно, поэтому с первого раза у него ничего не вышло. Тогда Вася заготовил ему еще несколько примеров, а для того, чтобы ему было проще проверять, попросил вас написать программу, которая по заданным треугольникам выводит ответ на задачу.

Формат входного файла

Первая строка входного файла содержит натуральное число n — количество треугольников ($1 \le n \le 50$). Следующие n строк содержат по шесть целых чисел — $x_1, y_1, x_2, y_2, x_3, y_3$, разделенных пробелами. Эти величины задают треугольник с вершинами в точках (x_1, y_1), (x_2, y_2) и (x_3, y_3). Все числа во входном файле не превышают по модулю 10^8.

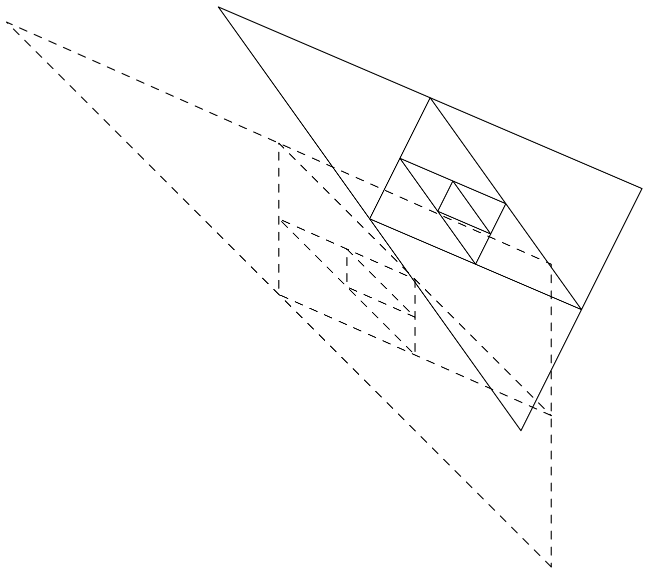
Формат выходного файла

В первой строке выходного файла выведите одно целое число — через какое число шагов найдется пара пересекающихся треугольников.

Примеры

triangles.in	triangles.out
2 0 0 1 0 0 1 0 0 -1 0 0 -1	0
2 0 0 1 0 0 1 -1 -1 -1 3 3 -1	0
2 10 2 5 9 3 5 -9 0 0 -9 -9 -5	3

Пояснение для последнего примера:



Сплошными линиями обозначены расширения первого треугольника, пунктирными — второго.

Задача E. Максимумы

Имя входного файла:	`peaks.in`
Имя выходного файла:	`peaks.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 megabytes

Говорят, что перестановка $\langle a_1, a_2, \ldots, a_n \rangle$ целых чисел от 1 до n имеет k максимумов, если неравенство $a_{i-1} < a_i > a_{i+1}$ выполняется ровно для k различных позиций i (будем считать, что $a_0 = a_{n+1} = 0$).

Например, у перестановки $\langle 3, 1, 4, 5, 2 \rangle$ два максимума: $i = 1$ и $i = 4$.

По заданным n и k найдите количество перестановок чисел от 1 до n ровно с k максимумами. Верните это число по модулю 239.

Формат входного файла

Входной файл содержит два целых числа: n и k ($1 \le n \le 10^{15}$, $1 \le k \le 30$).

Формат выходного файла

Выведите одно целое число — количество перестановок чисел от 1 до n ровно с k максимумами, взятое по модулю 239.

Примеры

peaks.in	peaks.out
3 1	4
10 3	131