

## Задача А. Соревнование эльфов

Имя входного файла: `compete.in`

Ограничение по времени: 1 секунда

Имя выходного файла: `compete.out`

Ограничение по памяти: 64 Мб

Каждый Новый год Санта Клаус устраивает международный чемпионат новогодних эльфов. Чемпионат этот проводится в два тура, в которых эльфы проявляют свое предновогоднее мастерство. В первом туре все эльфы разбиваются на команды по пять человек (точнее, эльфов) в каждой и играют в баскетбол. Во втором туре эльфы должны проявить себя настоящими предновогодними программистами. Для этого они разбиваются на команды по три эльфа и участвуют в IV открытом чемпионате Харькова по спортивному программированию.

Однако соревнование может быть поставлено под угрозу срыва, если количество эльфов-участников не будет удовлетворять определенным условиям. Действительно, ведь не всегда количество участвующих эльфов делится и на 3, и на 5. Поэтому Санта Клаусу нужно уметь определять, можно ли провести в этом году каждый из туров соревнований.

В этом году, организаторы IV открытом чемпионате Харькова по спортивному программированию попросили Санта Клауса помочь им в составлении условий задач. Хитрый Санта решил предложить им задачу про туры соревнований эльфов. Поэтому вам нужно определить, какие из туров соревнований можно провести, если в чемпионате участвует  $N$  эльфов. Поскольку Санте сказали, что IV чемпионат Харькова – это соревнование программистов, он решил усложнить задачу, задав  $N$  в двоичном виде.

### Входные данные

Во входном файле содержится единственное целое положительное число  $N$ , записанное в двоичной системе счисления без ведущих нулей. Число состоит из не более чем 10 000 двоичных разрядов.

### Выходные данные

Если Санта Клаус может провести оба тура соревнований, в единственную строку выходного файла выведите строку «BOTH», если только первый тур – строку «FIRST», если только второй – строку «SECOND», если же Санте не удастся провести ни одного тура – выведите строку «NONE». Все строки следует выводить большими буквами без кавычек.

### Пример

<code>compete.in</code>	<code>compete.out</code>
1111	BOTH

## Задача В. Кыш Бабай

Имя входного файла: babai.in

Ограничение по времени: 1 секунда

Имя выходного файла: babai.out

Ограничение по памяти: 64 Мб

Эх, тяжело быть Дедом Морозом в Татарстане. Еще бы, какому Деду Морозу понравится, когда местные жители называют его «Кыш Бабай»?! Но к этому можно привыкнуть, ведь даже если тебя и обзывают всякими "бабаями", то при этом ждут тебя в канун Нового Года ничуть не меньше.

Настоящая беда Деда Мороза в Татарстане (или правильнее, Кыш Бабая) в другом. По местным поверьям, год будет счастливым, если его номер делится нацело на 137 (как известно, это одна из основополагающих констант нашей Вселенной, определяющая многие процессы от формирования планет до квантовых эффектов). Но вот беда, такие счастливые года встречаются крайне редко. Поэтому Кыш Бабай, чтобы хоть как-то утешить татарстанцев, решил записывать на всех открытках и транспарантах номер года несколько раз подряд, чтобы получившееся число было счастливым, например была выпущена открытка "С новым 20092009 годом!", ведь число 20092009 делится на 137.

Идея Кыш Бабая заинтересовала Дедов Морозов других стран, но проблема в том, что в других странах верят в другие счастливые фундаментальные физические константы. Поэтому Кыш Бабай просит вас написать универсальную программу, которая быстро вычисляет, сколько раз нужно повторить номер года, чтобы он был счастливым, если известны номер этого года  $N$  и счастливая константа  $K$ .

### Входные данные

В единственной строке входного файла содержатся два целых числа  $N$  и  $K$  ( $1 \leq N \leq 1\,000\,000\,000$ ,  $1 \leq K \leq 100\,000$ ).

### Выходные данные

В выходной файл выведите повторенный несколько раз номер года  $N$ , чтобы он делился на фундаментальную константу. При этом добавлять ведущие нули не разрешается. Если существует несколько ответов, выведите кратчайший из них. Если достичь делимости никаким числом повторений нельзя, то выведите строку «NO NEW YEAR» (без кавычек).

### Пример

babai.in	babai.out
2009 137	20092009
2009 49	2009

## Задача С. Йоулупукки

Имя входного файла: joulu.in

Ограничение по времени: 1 секунда

Имя выходного файла: joulu.out

Ограничение по памяти: 64 Мб

Йоулупукки был приятно удивлен, когда вчера ему позвонил его старый друг, украинский коллега Миколай Бородатий<sup>©</sup> и пригласил поучаствовать в IV открытом чемпионате Харькова по спортивному программированию. Еще бы, не часто провинциальному финскому Деду Морозу выпадает шанс участвовать в соревновании столь высокого уровня.

Поэтому, еще за неделю до Чемпионата, Йоулупукки вместе со своей женой Муори вышел в интернет и стал разрабатывать маршрут поездки. Дело это непростое, ведь путь из Финляндии в Украину неблизкий и тяжелый, особенно зимой. Да и к тому же мировой экономический кризис отразился даже на Дедах Морозах, поэтому Йоулупукки хочет потратить как можно меньше денег на поездку.

Йоулупукки летает исключительно на санях, запряженных особыми летающими финскими лайками, которых предоставляет единственная авиасобакокомпания FinAirDogLines. На сайте FinAirDogLines Йоулупукки узнал, что в этом сезоне компания осуществляет  $M$  двухсторонних рейсов между  $N$  городами Европы. При этом резиденция Йоулупукки, гора Корватунтури, имеет номер 1, а международный аэрособакопорт Харькова – номер  $N$ .

Рейс номер  $i$  соединяет между собой города  $A_i$  и  $B_i$  и выполняется на санях, запряженных  $C_i$  собаками. Одна и та же собака может летать в нескольких рейсах. Стоимость аренды одной собаки составляет 1 евро на любое число рейсов.

Йоулупукки уже не первый год пользуется услугами FinAirDogLines и авиасобакокомпания согласна оплатить ему любые  $K$  перелетов, предоставляя своих собак абсолютно бесплатно, но только для этих  $K$  перелетов. Собак для остальных рейсов Йоулупукки должен арендовать.

К сожалению, Йоулупукки отличается сильной неторопливостью, присущей всем финнам, поэтому он точно не успеет за неделю вычислить количество собак, которых ему нужно арендовать для поездки из своей резиденции в Харьков. Помогите ему в этих подсчетах.

### Входные данные

Первая строка входного файла содержит 3 целых числа  $N$ ,  $M$  и  $K$  ( $2 \leq N \leq 1000$ ,  $1 \leq M \leq 10000$ ,  $0 \leq K < N$ ). Каждая из следующих  $M$  строк описывает один двухсторонний рейс и содержит 3 целых числа  $A_i$ ,  $B_i$ ,  $C_i$  ( $1 \leq A_i, B_i \leq N$ ,  $A_i \neq B_i$ ,  $1 \leq C_i \leq 10^9$ ). Никакая пара чисел  $(A_i, B_i)$  не встречается более одного раза.

### Выходные данные

В выходной файл выведите единственное число – минимальное количество собак, которое должен арендовать Йоулупукки, чтобы добраться из своей резиденции в Харьков. Если Йоулупукки не сможет добраться до Харькова, выведите «-1» (без кавычек).

### Пример

joulu.in	joulu.out	Примечание
5 7 1 1 2 5 3 1 4 2 4 8 3 2 3 5 2 9 3 4 7 4 5 6	4	Йоулупукки следует добираться по маршруту 1→3→2→5, при этом авиасобакокомпания бесплатно предоставляет полет между городами 2 и 5.

## Задача D. Шоколадная победа

Имя входного файла: `choco.in`

Ограничение по времени: 1 секунда

Имя выходного файла: `choco.out`

Ограничение по памяти: 64 Мб

Вот это победа! Весь мир потрясен! В канун Нового 2009 года на финале соревнований по спортивному программированию по системе ACM для Дедов Морозов впервые в истории победила сборная Латинской Америки – команда «DedosLatinoMorozos». В составе команды выступали Деды Морозы, или вернее Papá Noel-ы, трех стран – Бразилии, Аргентины и Чили, под руководством тренера, заслуженного Papá Noel-а из Венесуэлы.

Однако единству новогодней сборной южноамериканского континента был нанесен непоправимый удар организаторами. В качестве призов команде-победительнице подарили  $N$  новогодних шоколадок, весом от 1 до  $N$  килограмм. Все участники команды – Деды Морозы – очень темпераментные, никто из них не позволит себя обидеть при разделе подаренных шоколадок. Поэтому единственное решение – это поделить шоколадки между тремя членами команды и тренером так, чтобы каждый получил по равному количеству килограммов шоколада. Кроме того, шоколадки дороги всем членам команды как память о победе, поэтому они не согласны делить ни одну из шоколадок на части.

Такие требования победителей поставили в тупик организаторов Чемпионата, и они просят вас проверить, можно ли разделить  $N$  шоколадок указанным образом и если это возможно, построить один из вариантов дележа. Учтите, что участники команды могут получить разное количество шоколадок, главное, чтобы суммарный вес шоколадок у каждого участника совпадал.

### Входные данные

Входной файл содержит единственное целое число  $N$  ( $1 \leq N \leq 1000$ ) – количество призовых шоколадок.

### Выходные данные

Если поделить  $N$  шоколадок между участниками команды невозможно, выведите в выходной файл строку «NO CHOCO» (без кавычек). Если же решение существует, то выведите в выходной файл 4 строки, каждая из которых содержит несколько целых чисел, разделенных пробелом – веса шоколадок, которые должен получить соответствующий участник. Числа в строке можете выводить в любом порядке. Если вариантов разбиения несколько, выведите любой из них.

### Пример

<code>choco.in</code>	<code>choco.out</code>
15	7 5 6 8 4 2 12 3 13 9 11 10 1 14 15

## Задача E. Новогодний торт

Имя входного файла: `cake.in`

Ограничение по времени: 1 секунда

Имя выходного файла: `cake.out`

Ограничение по памяти: 64 Мб

Новый год отмечают все, в том числе и такие легендарные личности, как Вася Пупкин. Новый 2009 год Вася будет встречать дома, куда пригласит  $N$  своих лучших друзей. Для них он решил приготовить свой знаменитый новогодний торт. По замыслу Васи, торт должен иметь форму выпуклого  $(N+3)$ -угольника. Тогда Вася будет резать торт следующим хитрым образом. Каждый раз он будет выбирать три последовательные вершины многоугольника и одиночным разрезом отрезать получившийся треугольный кусок торта очередному гостю. В результате после  $N$  разрезов у Васи останется его собственный кусок, также имеющий форму треугольника.

Однако Вася очень любит новогодние торты, поэтому он решил таким образом отрезать от торта куски, чтобы его собственный последний кусок был максимального размера. Для этого он быстренько составил программу, которая посчитала площадь Васиного куска торта. А сможете ли вы написать такую программу?

### Входные данные

В первой строке входного файла записано целое число  $N$  ( $0 \leq N \leq 50$ ). Каждая из следующих  $(N+3)$  строк содержит два целых числа  $X$  и  $Y$ , по модулю не превосходящих 10000 – координаты вершин многоугольника в некоторой декартовой системе координат в порядке обхода по часовой стрелке.

### Выходные данные

В единственную строку выходного файла выведите максимально возможную площадь Васиного куска торта. Ответ будет засчитан, если он равен правильному результату с точностью до 0.001.

### Пример

<code>cake.in</code>	<code>cake.out</code>
1 1 1 1 2 3 3 2 1	1.5

## Задача F. Иллюминация

Имя входного файла: `light.in`

Ограничение по времени: 1 секунда

Имя выходного файла: `light.out`

Ограничение по памяти: 64 Мб

На международном 2009-ом съезде Дедов Морозов, который проходил в начале декабря этого года в Болгарии под руководством болгарского Деда Мороза Дяди Коледы, рассматривались важнейшие вопросы организации предстоящего новогоднего праздника. Учитывая мировой экономический спад и динамику цен на энергоносители, Деда Морозы рассматривали вопрос подсветки елок, установленных на центральных площадях каждого из  $N$  городов Восточной Европы. Было решено не использовать местную электросеть для подсветки, а создать отдельную систему для снабжения новогодних елок электричеством.

В каждом из городов можно построить специальную атомную электростанцию для генерации новогоднего электричества. Учитывая разную экологическую ситуацию в городах, стоимость постройки электростанций различна для разных городов, а именно составляет  $C_i$  евроугриков для  $i$ -го города. Вместо строительства электростанции, город можно соединить кабелем с другим городом, куда уже было подведено новогоднее электричество. При этом город может снабжаться электростанцией не напрямую, а через несколько других городов. Стоимость прокладки кабеля между  $i$ -ым и  $j$ -ым городами составляет  $A_{ij}$  евроугриков.

Определите стоимость системы новогоднего электроснабжения елок, построенной Дедами Морозами, если известно, что они выбрали самый дешевый вариант.

### Входные данные

В первой строке входного файла содержится единственное целое число  $N$  ( $1 \leq N \leq 300$ ) – количество городов. Во второй строке записаны  $N$  целых чисел  $C_i$ , разделенных пробелами ( $1 \leq C_i \leq 100\,000$ ). Следующие  $(N-1)$  строк описывают стоимость прокладки кабелей между городами. А именно  $(i+2)$ -ая строка содержит  $(N-i)$  целых чисел  $A_{i,i+1}, A_{i,i+2}, \dots, A_{i,N}$  ( $1 \leq A_{ij} \leq 100\,000$ ).

### Выходные данные

В выходной файл выведите единственное целое число – минимально возможную стоимость постройки системы электроснабжения елок, вычисленную в евроугриках.

### Пример

<code>light.in</code>	<code>light.out</code>	Примечание
4 1 2 3 4 1 5 5 2 5 5	8	Оптимально будет построить электростанции в городах 1 и 4 и проложить кабеля между парами городов 1 и 2, 2 и 3.

## Задача G. Фрукты

Имя входного файла: `fruit.in`

Ограничение по времени: 1 секунда

Имя выходного файла: `fruit.out`

Ограничение по памяти: 64 Мб

Маленький узбекский Дед Морозик Куорбобо неожиданно вспомнил, что через час наступает Новый 2009 год, а он так и не приготовил никаких фруктов для праздничного стола. А ведь дома его ждет небольшая семья – любящая жена и 20 обожаемых детишек. Поэтому Куорбобо непременно должен собрать фрукты.

Домик Куорбобо находится посреди волшебного прямоугольного поля, протянувшегося на  $A$  метров с Востока на Запад и на  $B$  метров с Севера на Юг. Поле со всех сторон окружено фруктовыми деревьями, плодоносящими круглый год (не забывайте, что поле волшебное). А именно на Востоке поля растут мандарины, на Юге – апельсины, на Западе – яблоки, а на Севере – тыблоки (не удивляйтесь, это местный волшебный узбекский фрукт). Если считать, что Юго-Западный угол поля находится в начале координат, то дом Куорбобо находится в точке с координатами  $(X_1, Y_1)$ , а сам Куорбобо стоит в точке  $(X_0, Y_0)$ .

Поскольку семья у Куорбобо сравнительно маленькая, он хочет принести домой всего один мандарин, один апельсин, одно яблочко и одно тыблочко. Для этого ему нужно дойти до каждой из сторон поля, сорвать соответствующий фрукт и вернуться домой. При этом, если Куорбобо дойдет до угла поля, то он сможет сорвать сразу два фрукта.

Хоть зимы на волшебных полях Узбекистана и не очень холодные, Куорбобо все равно хочет попасть домой поскорей, а потому решил двигаться оптимальным образом. Помогите ему составить оптимальный маршрут и выведите его длину, а также координаты точек, где Куорбобо должен сорвать фрукты. Учтите, что Куорбобо может срывать фрукты в произвольном порядке, главное – сорвать по одному фрукту каждого вида.

### Входные данные

В единственной строке входного файла записаны 6 целых чисел  $A, B, X_0, Y_0, X_1, Y_1$  ( $2 \leq A, B \leq 200, 1 \leq X_0, X_1 \leq A-1, 1 \leq Y_0, Y_1 \leq B-1$ ). Координаты начального положения Куорбобо и его дома могут совпадать.

### Выходные данные

В первую строку выведите длину оптимального маршрута Куорбобо. В каждой из следующих 4 строк выведите по 2 числа – координаты точек, где Куорбобо должен срывать фрукты: в первой строке – координаты точки для мандарина, во второй – апельсина, в третьей – яблока, в четвертой – тыблока. Все числа следует выводить с максимальной точностью. Значения будут сравниваться с точностью до 0.00001. Если существует несколько решений с минимальной длиной, выведите любое из них.

### Пример

<code>fruit.in</code>	<code>fruit.out</code>	Примечание
4 3 1 1 3 2	7.810249676 4.000000000 2.833333333 0.200000000 0.000000000 0.000000000 0.000000000 0.166666667 3.800000000 3.000000000	

## Задача Н. Вселенский Новый год

Имя входного файла: `universe.in`

Ограничение по времени: 1 секунда

Имя выходного файла: `universe.out`

Ограничение по памяти: 64 Мб

На первый взгляд кажется, что Новый год – праздник исключительно земной, неразрывно связанный с вращением нашей планеты вокруг Солнца. На самом деле светлых и позитивных праздников хочется всем, поэтому аналоги Нового года отмечаются во всех уголках Вселенной. И все же, не будем замахиваться на масштабы Вселенной, а ограничимся окрестностями нашей звезды – Солнечной системой.

В Солнечной системе тоже отмечается Новый год. Вот только поскольку Солнце немного больше Земли, то и Новый год в масштабах Солнечной системы наступает значительно реже. Существует множество трактовок «года» в рамках Солнечной системы, но мы рассмотрим трактовку на основе трех самых близких нам планет – Венеры, Земли и Марса. Еще из древнеиндийской астрономии известно, что в Новый год обнуляются все астрономические переменные. В качестве таких переменных в нашей модели возьмем угловое положение Венеры, Земли и Марса.

Период вращения Венеры вокруг Солнца равен 5293 часам, Земли – 8766 часам, Марса – 16487 часам. Поэтому Новый год в Солнечной системе наступает каждые 764971047306 часов (т.е. приблизительно каждые 87 миллионов лет).

Ясное дело, что Новый год Солнечной системы – это огромный праздник для всех Дедов Морозов планеты. Поэтому они обратились к вам с просьбой определить, сколько еще часов осталось им ждать Нового года Солнечной системы. Специально для вас Деда Морозы узнали у знакомых астрономов, сколько часов назад каждая из планет проходила через точку с нулевым угловым положением. К сожалению, астрономы уже готовились к предстоящему земному Новому году, поэтому дали несколько различных наборов значений. Деда Морозы для надежности просят вас определить оставшееся время для каждого набора текущих положений планет.

### Входные данные

В первой строке записано единственное целое число  $N$  ( $1 \leq N \leq 10\,000$ ) – количество наборов входных данных. Каждая из следующих  $N$  строк описывает один набор положений планет и содержит 3 целых числа  $V, E, M$  ( $0 \leq V < 5293, 0 \leq E < 8766, 0 \leq M < 16487, V+E+M > 0$ ) – время в часах, прошедшее с момента последнего обнуления углового положения Венеры, Земли и Марса соответственно.

### Выходные данные

В выходной файл выведите  $N$  строк, каждая из которых содержит время в часах до следующего обнуления положений планет для соответствующего текущего положения планет.

### Пример

<code>universe.in</code>	<code>universe.out</code>
2	764971047305
1 1 1	3
5290 8763 16484	



## Задача I. Гирлянда

Имя входного файла: `lamps.in`

Ограничение по времени: 3 секунды

Имя выходного файла: `lamps.out`

Ограничение по памяти: 64 Мб

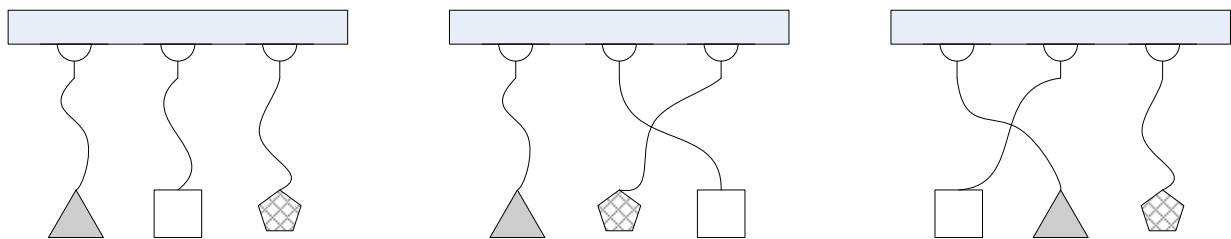
По личному распоряжению Президента республики Беларусь, 31 декабря 2008 года все население республики должно быть готово к встрече Нового 2009 года. Распоряжение это касается всех жителей Беларуси, в том числе и белорусского коллеги нашего Святого Миколая, Дзеда Мароза, также известного как Зюзя.

Зюзя очень любит Новый год и еще больше уважает своего президента, поэтому он решил подготовиться к празднику основательно. Установив на главной площади елку, купив украшений, засыпав всю республику снегом и подарив всем ее жителям новогоднее настроение, наш Дзед Мароз понял, что ему чего не хватает. А не хватало ему гирлянды. Да, да, длинной мигающей гирлянды, состоящей из множества разноцветных лампочек.

К счастью, как раз в канун Нового года отечественный производитель гирлянд – Белорусский Национальный Гирляндо-паятельный завод изготовил опытный образец гирлянды нового поколения и предоставил его Дзеду Морозу для тестирования.

Новая модель гирлянды состоит из опорной планки и  $N$  лампочек различных цветов, расположенных в определенном порядке. Каждая лампочка крепится к опорной планке проводом длиной  $K$  метров. При этом точки закрепления проводов расположены в ряд с расстоянием 1 метр между соседними. Благодаря такой прогрессивной конструкции, Зюзя может менять порядок следования лампочек в гирлянде, переставляя соседние лампочки, пока это позволяет длина подключенных к ним проводов. Т.е. каждая лампочка может быть перемещена в любую сторону не более чем на  $K$  метров от своего начального положения.

Например, для гирлянды, состоящей из 3 лампочек, соединенных проводами в 1 метр, Зюзя может получить 3 возможных расположения лампочек:



Помогите Дзеду Морозу определить, сколькими способами он может расставить лампочки в гирлянде.

### Входные данные

В единственной строке входного файла записаны два целых числа  $N$  и  $K$  ( $1 \leq N \leq 100$ ,  $1 \leq K \leq 6$ ).

### Выходные данные

В выходной файл выведите единственное число – количество способов расставить лампочки в гирлянде.

### Пример

<code>lamps.in</code>	<code>lamps.out</code>
3 1	3

## Задача J. Подарки

Имя входного файла: `gift.in`

Ограничение по времени: 1 секунда

Имя выходного файла: `gift.out`

Ограничение по памяти: 64 Мб

Как и все жители его страны, немецкий Дед Мороз *Weihnachtsmann* отличался исконно немецкой педантичностью и ответственностью. Поэтому он решил заранее разложить все новогодние подарки детишкам по порядку, в отличие от своих славянских коллег, которые обычно просто скидывают все подарки в один громадный мешок. *Weihnachtsmann* решил использовать в качестве склада для подарков свою кладовую, имеющие размеры  $N$  на  $M$  метров. Но поскольку немецкому Деду Морозу нужно было отвечать на письма детишек, он поручил работу по укладыванию  $N \cdot M$  подарков в кладовой своему помощнику, гномику Руби.

Руби подошел к проблеме основательно. Для начала он ввел в кладовой декартову систему координат, так что оси параллельны сторонам кладовой, центр координат находится в одном из углов кладовой, а вся кладовая лежит в первом квадранте, так что ее размер по оси  $X$  равен  $N$  метров. В таком случае кладовая оказалась условно разбита на  $N \cdot M$  единичных квадратиков. В каждый такой квадратик Руби должен положить по одному подарку.

Беда в том, что Руби уже достаточно стар и плохо видит, а в кладовой плохое освещение. Поэтому он разработал специальный алгоритм укладки подарков «на ощупь». Руби начинает с клетки, расположенной в начале координат, и двигается в положительном направлении оси  $OX$ . Каждый раз, когда Руби упирается в стенку или в клетку с уже установленным подарком, он поворачивает влево на  $90$  градусов. Проходя через очередную клетку, он укладывает в ней следующий по порядку подарок. Таким образом, Руби проходит через каждую клетку ровно один раз. Уложив таким способом все подарки, Руби с довольным видом отправился в пивную выпить кружечку настоящего немецкого пива.

В это время *Weihnachtsmann* закончил разбор корреспонденции и отправился поглядеть на результаты трудов Руби. Немецкого Деда Мороза заинтересовал вопрос, сможете ли вы по координатам клетки определить, кому предназначается находящийся в ней подарок, если известно, что Руби чередовал подарки в следующем порядке: первый – для мальчика, второй – для девочки, третий – для гномика, четвертый – для мальчика, и т.д.

### Входные данные

Первая строка входного файла содержит три целых числа  $N$ ,  $M$ ,  $K$  ( $1 \leq N, M \leq 10^9$ ,  $N \cdot M \leq 10^9$ ,  $1 \leq K \leq 50\,000$ ) – размеры кладовой и количество клеток, для которых нужно определить тип подарка. В следующих  $K$  строках записано по два числа  $X$  и  $Y$  ( $1 \leq X \leq N$ ,  $1 \leq Y \leq M$ ) – координаты каждой из этих клеток.

### Выходные данные

В выходной файл выведите  $K$  строк, каждая из которых содержит одно из слов «BOY», «GIRL» или «GNOME» (без кавычек), в зависимости от того, кому предназначается подарок в соответствующей клетке.

### Пример

<code>gift.in</code>	<code>gift.out</code>
4 3 6	BOY
1 1	GIRL
2 1	GNOME
3 1	BOY
4 1	GIRL
2 2	GNOME
3 2	