

## Задача А. Дырки и картины

Имя входного файла: `pictures.in`  
Имя выходного файла: `pictures.out`  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

– ...Вот какая польза от этой картины на стене?  
– От этой картины на стене, – говорит мама, – очень большая польза. Она дырку на обоях загоразживает.

Э.Успенский «Дядя Федор, пес и кот», глава 1

В доказательство своих слов мама решила привести строгий расчет пользы от картин, висящих в гостиной. Вооружив папу рулеткой, мама составила подробную схему имеющихся в обоях дырок и висящих на стене картин.

Оказалось, что в обоях имеется  $N$  прямоугольных дырок, причем стороны дырок расположены строго горизонтально и вертикально. Никакие две дырки не пересекаются, но могут касаться сторонами.

Поверх обоев на стене висит  $M$  прямоугольных картин, со сторонами, расположенными строго вертикально и горизонтально. Естественно, никакие две картины не пересекаются, но могут касаться сторонами.

Мама Римма просит вас подсчитать, какова площадь частей дырок в обоях, которые не закрыты картинами.

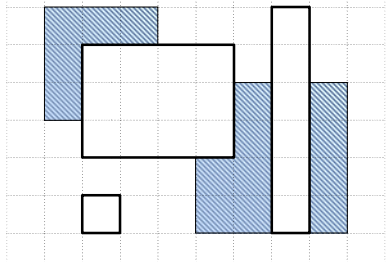
### Входные данные

Первая строка входного файла содержит единственное целое число  $N$  ( $0 \leq N \leq 1000$ ) – количество дырок в обоях. Следующие  $N$  строк содержат описания дырок по одной в строке. Каждая дырка задается четырьмя целыми числами  $X_1, Y_1, X_2, Y_2$  ( $-10^9 \leq X_1 < X_2 \leq 10^9, -10^9 \leq Y_1 < Y_2 \leq 10^9$ ) – координатами противоположных вершин дырки-прямоугольника в декартовой системе координат, оси которой расположены горизонтально и вертикально. Последующая строка содержит единственное целое число  $M$  ( $0 \leq M \leq 1000$ ) – количество картин, висящих на стене. Следующие  $M$  строк содержат описания картин в формате, аналогичном описанию дырок.

### Выходные данные

В выходной файл выведите единственное целое число – площадь частей дырок в обоях, не закрытых картинами.

### Пример

| <code>pictures.in</code>                                      | <code>pictures.out</code> | Примечание   |
|---|---------------------------|--|
| 2<br>1 4 4 7<br>5 1 9 5<br>3<br>2 3 6 6<br>2 1 3 2<br>7 1 8 7 | 15                        | Дырки и картины расположены, как на рисунке:<br> |

## Задача В. Почта

Имя входного файла: `post.in`  
Имя выходного файла: `post.out`  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

*Шарик достал уголёк и на боку печки стал рисовать домик.*

*– Эй, - спрашивает кот, – что это? Что это за народное творчество на моей печке?*

*– Это индейская национальная изба, – ехидно отвечает Шарик, – "фигвам" называется.*

*– Что? – кричит Матроскин. – Да я в него за это утюгом! Где мой утюг деревенский с углями?*

*Он притащил огромный чугунный утюг, прямо как из музея.*

*– Стоп-стоп! – говорит Печкин. – Это уже посылка получается. Платите двадцать рублей за доставку. Я уж ваш утюг передам.*

*Э.Успенский «Тетья дяди Федора», глава 9*

В результате ссоры по политическим причинам кот Матроскин и пес Шарик уже целый час не разговаривают. Поэтому свои политические споры они продолжают с помощью почтальона Печкина. Причем наши герои очень быстро перешли от словесных прений к вполне материальным методам убеждения, вроде кочерги и утюга. Но вот беда: Матроскин с Шариком все-таки не разговаривают. Поэтому им приходится пользоваться почтой для передачи друг другу различных увесистых «аргументов».

Но прежде чем запустить (а точнее переслать почтой) в Шарика утюгом, экономный кот Матроскин призадумался. «Это что ж получается», – рассуждал он – «я на этого балбеса еще и деньги тратить должен! Если уж и так, то надо постараться потратить как можно меньше».

Кот хочет переслать  $N$  предметов, причем непременно в указанном порядке (на то есть очень важные политические причины). Коту в точности известен вес каждого из предметов в граммах. Почтальон Печкин может передавать предметы только письмом, бандеролью или посылкой. Стоимость письма составляет  $C_1$  рублей, а максимальный вес –  $G_1$  граммов. Для бандероли –  $C_2$  рублей и  $G_2$  граммов, а для посылки –  $C_3$  рублей и  $G_3$  граммов. Помогите коту Матроскину переслать все предметы в заданном порядке Шарiku, потратив при этом как можно меньше денег.

### Входные данные

Первая строка входного файла содержит единственное натуральное число  $N$  ( $1 \leq N \leq 10^6$ ) – количество предметов. Во второй строке перечислены  $N$  целых чисел  $A_i$  ( $1 \leq A_i \leq G_3$ ) – веса пересылаемых предметов в порядке их пересылки. Третья строка содержит 3 натуральных числа  $0 < C_1 < C_2 < C_3 \leq 10^9$  – стоимости письма, бандероли и посылки соответственно. Четвертая строка также содержит 3 натуральных числа  $0 < G_1 < G_2 < G_3 \leq 10^9$  – максимальный вес письма, бандероли и посылки соответственно в граммах.

### Выходные данные

В выходной файл выведите единственное целое число – минимально возможную стоимость отправки всех предметов в заданном порядке Шарiku.

### Пример

| <code>post.in</code> | <code>post.out</code> |
|----------------------|-----------------------|
| 3                    | 80                    |
| 12 998 20            |                       |
| 15 20 50             |                       |
| 20 40 1000           |                       |

## Задача С. Галчонок Хватайка на вишне

Имя входного файла: cherry.in  
Имя выходного файла: cherry.out  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

Возле дома дяди Федора росла огромная-преогромная вишня. И каждый год она давала огромный урожай. Вот и в этом году дядя Федор, кот Матроскин и Шарик наелись вишен до отвала. Но сезон закончился, дерево опустело, и на нем осталась всего одна вишенка.

Заметим, что, как и любое дерево, вишня представляет собой граф с  $N$  вершинами и  $N-1$  ребром. Вершина под номером 1 является корнем дерева, а в вершине под номером  $K$  висит единственная вишенка.

Однажды галчонок Хватайка решил съесть эту одинокую вишенку. Но поскольку делать ему было нечего, то он решил немного развлечься. Изначально галчонок сел в корень (т.е. в вершину 1) дерева. Каждую минуту галчонок перелетает из текущей вершины в любую из соседних, т.е. таких, которые соединены с текущей вершиной веткой (т.е. ребром). Причем вершину, куда перелететь, он выбирает случайно, равновероятным образом. Когда галчонок оказывается в вершине, где растет вишенка, он ее съедает и с чувством выполненного долга летит отдыхать.

За более чем интересной игрой галчонка Хватайки внимательно наблюдал Шарик, прилегший отдохнуть под соседней яблоней. Дело в том, что Шарик недавно прочитал некоторые книги по теории вероятностей и теперь его очень интересовал вопрос: чему же равно математическое ожидание времени, которое пройдет до момента, когда галчонок все-таки съест вишенку. На этот интересный вопрос Шарик просит ответить вас.

### Входные данные

Первая строка входного файла содержит два натуральных числа – количество вершин в дереве  $N$  ( $1 \leq N \leq 50$ ) и номер вершины, где растет вишенка  $K$  ( $1 \leq K \leq N$ ). Каждая из следующих  $N-1$  строк описывает ветку дерева. Каждая такая строка содержит два натуральных числа  $X$  и  $Y$  ( $1 \leq X < Y \leq N$ ), которые означают, что вершины  $X$  и  $Y$  соединяет ветка.

### Выходные данные

В выходной файл выведите единственное вещественное число с четырьмя знаками после запятой – математическое ожидание времени (в минутах), которое потребуется галчонку, чтобы добраться до вишенки.

### Пример

| cherry.in                | cherry.out |
|--------------------------|------------|
| 4 3<br>1 2<br>2 3<br>2 4 | 6.0000     |

## Задача D. Настольная игра

Имя входного файла: `game.in`  
Имя выходного файла: `game.out`  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

Долгими зимними вечерами Матроскин и Шарик очень любят играть в различные настольные игры. Вы ведь помните, какие это игры? В обычной настольной игре имеется поле, на котором нарисованы  $N$  ячеек, последовательно пронумерованных от 1 до  $N$ . Игра ведется фишкой, которая установлена на клетку 1. По правилам, игроки по очереди бросают кубик и делают ходы. Выигрывает игрок, который делает ход дальше  $N$ -ой клетки. Кроме того, на игровом поле нарисованы  $M$  стрелок. Каждая стрелка идет от некоторой клетки  $A$  до клетки  $B$ . Если игрок заканчивает свой ход в клетке  $A$ , то фишка немедленно переносится в клетку  $B$ . В несложных играх, которые рассматриваются в этой задаче, все стрелки ведут вперед, т.е.  $B > A$  и из каждой клетки выходит не более одной стрелки. Также никакая стрелка не начинается в клетке, где заканчивается другая стрелка.

Но вся проблема состоит в том, что Шарик во время последней уборки потерял игровой кубик. Поэтому играть по правилам никак не получится. «А давайте мы немного изменим правила», – предложил кот Матроскин. – «Пусть вместо броска кубика, игрок сам определяет, на сколько клеточек вперед он продвинет фишку. И пусть он может продвинуть фишку только на 2, 3 или 5 клеточек. Тогда и кубик не понадобится, и игра станет интеллектуальной и независимой от воли случая». Все очень обрадовались такому изменению правил игры, ведь старые правила уже порядком поднадоели.

Но Шарик заметил, что у новых правил есть существенный недостаток – перед тем как сделать ход, нужно изрядно подумать, как же выиграть. Поэтому Шарик просит вас написать ему программу, которая по заданному игровому полю определяла бы, какой игрок побеждает при правильной игре. Дядю Федора, наблюдающего за игрой, также интересует, сколько продлится партия. Будем считать, что и Матроскин и Шарик играют оптимально, причем игрок, который выигрывает, пытается выиграть как можно быстрее, а который проигрывает – наоборот, затягивает партию.

### Входные данные

Первая строка входного файла содержит два целых числа  $N$  и  $M$  ( $1 \leq N \leq 10000$ ,  $0 \leq M \leq N/2$ ). Каждая из следующих  $M$  строк описывает стрелку и содержит два натуральных числа  $A_i$ ,  $B_i$  ( $2 \leq A_i < B_i \leq N$ ). Все числа  $A_i$  различны.

### Выходные данные

Если игрок, который первым делает ход, побеждает при правильной игре, то в первую строку выходного файла выведите слово “FIRST” (без кавычек). Если же побеждает второй игрок, то слово “SECOND”. Во второй строке выведите единственное целое число – количество ходов в партии при оптимальной игре.

### Пример

| game.in           | game.out    | Примечание  |
|-------------------|-------------|---|
| 6 2<br>3 6<br>4 6 | SECOND<br>2 | Своим ходом первый игрок может попасть в клетки 3, 4 или 6. Учитывая стрелки, после его хода фишка окажется в клетке 6. После чего второму игроку будет достаточно сделать любой ход. |
| 2 0               | FIRST<br>1  | Даже шагнув на две клеточки, первый игрок попадает дальше $N$ -ой клетки и побеждает.   |

## Задача Е. Еще одна игра

Имя входного файла: `game2.in`  
Имя выходного файла: `game2.out`  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

*Тут в дверь постучали. Хватайка спрашивает:*

*- Кто там?*

*Из-за двери отвечают:*

*- Свои.*

*Кот говорит:*

*- В такую погоду свои дома сидят. Телевизор смотрят. Только чужие шастают. Не будем дверь открывать!*

*Э.Успенский «Дядя Федор, пес и кот», глава 21*

После того, как Шарик обзавелся программой для игры в настольные игры со стрелками, соревноваться с ним стало абсолютно бесполезно. Ведь он сразу же может сказать, кто выиграет в партии. Но до конца зимы оставалось еще больше месяца, а потому нужно было коротать долгие вечера за каким-нибудь занятием. Поэтому дядя Федор предложил Матроскину и Шарик у новую игру.

Новая игра – исключительно интеллектуальная и производится над заранее заданным числом  $N$ . Матроскин и Шарик по очереди делают ходы. За один ход игрок может некоторое ненулевое количество раз поделить или умножить данное число на любое простое число, при условии, что результат будет выражаться целым числом. Проигрывает тот, кто не может сделать ход. Так как при таких правилах игра может вообще не закончиться, Матроскин и Шарик договорились, что количество ходов для каждого игрока, в которых число было умножено, должно быть не больше  $K$ .

Хитрый Шарик опять просит вас определить кто выиграет при оптимальной игре обоих игроков.

### Входные данные

Единственная строка входного файла содержит два целых числа  $N$  и  $K$  ( $1 \leq N \leq 10^{12}$ ,  $0 \leq K \leq 10^{12}$ ).

### Выходные данные

Если игрок, который первым делает ход, побеждает при правильной игре, то в единственную строку выходного файла выведите 1. Если же побеждает второй игрок, то выведите 2.

### Пример

| <code>game2.in</code> | <code>game2.out</code> |
|-----------------------|------------------------|
| 18 1                  | 1                      |

## Задача F. Видеомагнитофон

Имя входного файла: vhs.in  
Имя выходного файла: vhs.out  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

*Ничего себе, вашу маму и там и тут передают, до чего техника дошла!*

*Из мультфильма «Зима в Простоквашино»*

Почтальон Печкин всегда очень любил различные технические устройства. После того, как он получил желанный велосипед, Печкин решил купить себе видеомагнитофон, чтобы записывать сериалы, идущие в одно время по разным каналам. Сказано – сделано, и почтальон приобрел староватый, но надежный видеомагнитофон VHS.

К магнитофону прилагалась одна кассета на  $L$  минут. А поскольку в Простоквашино все уже перешли на DVD, еще одну VHS кассету купить не так-то просто. Поэтому Печкин может записывать передачи только на одну кассету, которая шла в комплекте с видеомагнитофоном.

Существенный недостаток всех кассетных видеомагнитофоном состоит в том, что видео записывается последовательно на пленке. Поэтому при записи нужно быть осторожным, чтобы не стереть ранее записанные фрагменты. Такая проблема стоит и перед Печкиным. Он уже записал на кассете  $N$  непрерывных фрагментов. Теперь он хочет разместить на кассете новую серию сериала так, чтобы не повредить ни одного из уже записанных фрагментов.

Серия сериала представляет собой непрерывный блок длительностью  $D_1$  минут, рекламу длиной  $R$  минут и еще один непрерывный блок длиной  $D_2$  минут. Изначально кассета перемотана на начало. Почтальон Печкин до начала сериала может перемотать кассету в любое место и начать запись. В течение рекламы почтальон может перематывать кассету в любом направлении, чтобы после рекламы записать второй блок. Видеомагнитофон Печкина перематывает кассету в любую сторону со скоростью в  $V$  раз большей, чем скорость записи или воспроизведения.

Почтальон Печкин очень ценит свое время и поэтому хочет, чтобы суммарное время перемотки (до начала сериала и во время рекламы) было минимально возможным. Подскажите почтальону, как следует записывать сериал, или скажите, что это невозможно.

### Входные данные

Первая строка входного файла содержит пять целых чисел:  $N, L, V, D_1, R, D_2$  ( $0 \leq N \leq 1000, 1 \leq L, V, D_1, R, D_2 \leq 10^9$ ). Каждая из следующих  $N$  строк описывает один из уже записанных фрагментов и содержит два целых числа  $A_i$  и  $B_i$  ( $0 \leq A_i < B_i \leq L$ ) – метки начала и конца фрагмента. Метка – это число минут от начала кассеты, может быть от 0 до  $L$  включительно. Приведенные фрагменты не имеют общих точек и перечислены в произвольном порядке.

### Выходные данные

Если записать серию сериала можно, то выведите в выходной файл единственное вещественное число с шестью знаками после запятой – наименьшее суммарное время перемотки кассеты. Если записать серию не получится, то выведите слово “IMPOSSIBLE” (без кавычек).

### Пример

| vhs.in       | vhs.out    |
|--------------|------------|
| 0 20 1 5 5 5 | 0          |
| 1 20 1 5 5 5 | IMPOSSIBLE |
| 3 14         |            |

## Задача G. Платья

Имя входного файла: `dress.in`  
Имя выходного файла: `dress.out`  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

*Хватит, хватит, надоело! Моя жизнь совершенно беспросветная, я живу как крестьянка крепостная. У меня четыре платья есть вечерних, шелковых, а одеть их некуда. Нет, все решено - мы завтра же уезжаем на курорт.*

*Из мультфильма «Каникулы в Простоквашино»*

И мама Римма, твердо намеренная поехать завтра же на курорт, отправилась собирать чемоданы. А как известно, сборы на курорт – это очень важная и сложная задача, которую не решить без применения информационных технологий. Ведь взять хочется как можно больше разных вещей, а чемоданы не резиновые. Поэтому мама сразу же поручила дяде Федору разработать программу для вычисления оптимального состава багажа.

В гардеробе мамы есть  $A$  платьев,  $B$  шляпок и  $C$  сумочек, но в чемодан помещается не более  $N$  вещей. Мама хочет так выбрать одежду для отдыха, чтобы можно было совершить как можно больше светских выходов. На каждый светский выход мама будет надевать наряд, состоящий из платья, шляпки и сумочки, причем по правилам этикета, наряды, одетые на любые два светских раута, должны отличаться хотя бы двумя вещами. Помогите дяде Федору определить, какое максимальное количество светских выходов сможет совершить его мама в отпуске при оптимальной упаковке чемодана.

### Входные данные

Единственная строка входного файла содержит четыре целых числа  $A$ ,  $B$ ,  $C$  и  $N$  ( $0 \leq A, B, C, N \leq 10000$ ).

### Выходные данные

В выходной файл выведите единственное целое число – максимально возможное количество нарядов, различающихся хотя бы двумя вещами, которые можно составить при оптимальном выборе багажа.

### Пример

| <code>dress.in</code> | <code>dress.out</code> |
|-----------------------|------------------------|
| 10 10 10 5            | 2                      |
| 1000 1 1 1000         | 1                      |

## Задача Н. Фотоохота

Имя входного файла: photo.in  
Имя выходного файла: photo.out  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

*От Шарика тоже толку мало было. Потому что ему фоторужье купили. Он с утра в лес и полдня за зайцем носится, чтобы сфотографировать. А потом снова полдня за ним гоняется, чтобы фотографию отдать.*

*Э.Успенский «Дядя Федор, пес и кот», глава 16*

Шарику уже порядком надоело носиться за животными с фоторужьем. Поэтому Шарик решил кардинально поменять тактику – стоять на месте и ждать, пока заяц сам пробежит возле него. Для удачной фотографии нужно, чтобы заяц пробежал достаточно близко от Шарика, а именно на расстоянии не более чем  $R$  метров. Сможете ли вы определить, сколько придется ждать Шарика зайца?

Заяц начинает с точки  $(X_0, Y_0)$  и движется по ломаной  $(X_0, Y_0), (X_1, Y_1), \dots, (X_N, Y_N)$ . На прямолинейных участках он бежит с постоянной скоростью  $V$  метров в секунду. На поворот на угол  $\alpha$  заяц затрачивает  $K \cdot \alpha$  секунд. Шарик располагается в точке с координатами  $(X_{III}, Y_{III})$ . Определите, через сколько секунд после начала движения заяц окажется на расстоянии не больше чем  $R$  метров от Шарика.

### Входные данные

Первая строка входного файла содержит натуральное число  $N$  ( $2 \leq N \leq 5000$ ) и три вещественных числа  $R, V$  и  $K$  ( $0.01 \leq R \leq 10^6$ ;  $0.01 \leq K, V \leq 1000$ ). Вторая строка содержит два вещественных числа  $X_{III}$  и  $Y_{III}$  – координаты Шарика. Следующие  $N+1$  строк содержат по два вещественных числа  $X_i, Y_i$  в каждой – координаты очередной вершины ломаной. Все координаты по модулю не превосходят  $10^6$ .

### Выходные данные

В единственную строку выходного файла выведите единственное число с четырьмя знаками после запятой – время в секундах, прошедшее от начала движения до момента, когда Шарик сфотографирует зайца. Если Шарика так и не удастся это сделать, то выведите слово “NOPHOTO” (без кавычек).

### Пример

| photo.in     | photo.out |
|--------------|-----------|
| 3 1.5 1 0.01 | 2.4000    |
| 2 1          |           |
| 0 0          |           |
| 0 1          |           |
| 1 1          |           |



## Задача I. Дороги

Имя входного файла: `roads.in`  
Имя выходного файла: `roads.out`  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

*В связи с тем, что в деревне Простоквашино построено две гостиницы, аэродром, три военных санатория и проведена автотрасса российского значения, переименовать деревню Простоквашино в город Простоквашинск.*

*Мэром города назначить Ломовую-Бамбино Тамару Семёновну.*

*Э.Успенский «Дядя Федор, пес и кот», почти конец*

Вскоре Простоквашино (а точнее Простоквашинск) еще больше разрослось. Сегодня Простоквашинск – уже центр Простоквашинской области, а на должности мэра по-прежнему работает тетя дяди Федора, Тамара Семёновна. Одной из приоритетных задач развития области является постройка дорог.

При мэре Ломовой-Бамбино все деревни Простоквашинской области были соединены сетью высококлассных дорог так, что в каждую деревню можно проехать, причем единственным образом. Более того, за счет стандартизации строительства и переноса нескольких населенных пунктов удалось достичь ситуации, когда длина любой дороги, соединяющей два соседних населенных пункта, в точности равна 1 километру. Поэтому теперь расстояние между двумя населенными пунктами просто равно количеству дорог между ними.

Ввиду информатизации всех сфер управленческой деятельности было решено разработать программу, которая бы **быстро** могла вычислять расстояние между двумя населенными пунктами. Вам поручается разработать такую программу.

### Входные данные

Первая строка входного файла содержит единственное натуральное число  $N$  ( $1 \leq N \leq 15000$ ) – количество населенных пунктов (т.е. деревень и самого Простоквашинска) в области. Все населенные пункты пронумерованы от 1 до  $N$ , причем Простоквашинск имеет номер 1. Следующие  $N-1$  строк содержат по два различных натуральных числа  $A$  и  $B$  ( $1 \leq A, B \leq N$ ) – номера населенных пунктов, соединенных дорогой.

Следующая строка содержит единственное натуральное число  $M$  ( $1 \leq M \leq 30000$ ) – количество запросов к программе. Каждая из следующих  $M$  строк описывает запрос и содержит два натуральных числа  $A$  и  $B$  ( $1 \leq A, B \leq N$ ).

### Выходные данные

Для каждого запроса во входном файле выведите в выходной файл строку, содержащую ответ на запрос – расстояние между населенными пунктами  $A$  и  $B$ .

**Пример**

| roads.in | roads.out |
|----------|-----------|
| 8        | 4         |
| 1 2      | 2         |
| 2 3      | 2         |
| 3 4      | 0         |
| 3 5      | 3         |
| 3 6      |           |
| 6 7      |           |
| 7 8      |           |
| 5        |           |
| 2 8      |           |
| 4 6      |           |
| 1 3      |           |
| 5 5      |           |
| 7 4      |           |

**Подсказка**

Ваше решение должно быть **действительно быстрым**, чтобы пройти ограничение по времени на максимальном тесте.

## Задача J. Тетя дяди Федора

Имя входного файла: `army.in`  
Имя выходного файла: `army.out`  
Ограничение по времени: 2 сек  
Ограничение по памяти: 64 Мб

Попивая вечерами чай на веранде, тетя дяди Федора очень любила рассказывать интересные и поучительные истории из военной жизни. Однажды она поведала простоквашенским такую историю.

«Был у нас в роте прапорщик Шматко. И очень любил этот прапорщик отправлять солдат в наряд. Для выбора солдат в наряд прапорщик использовал очень хитрый способ, подсказанный одним математиком, которого с мехмата прямо армию забрали. Этот способ состоит в следующем.

Прапорщик составил список из  $N$  фамилий всех солдат роты, отсортированный по алфавиту. Каждый день прапорщик отправляет в наряд солдат, стоящих в списке под простыми номерами. При этом он вычеркивает их из списка и дописывает в начало списка, соблюдая относительный порядок.

Например, если изначально список состоял из 6 солдат под номерами: 1, 2, 3, 4, 5, 6, то после первого дня он будет выглядеть так: 2, 3, 5, 1, 4, 6, после второго дня так: 3, 5, 4, 2, 1, 6 и так далее.

Способ выбора солдат прапорщика Шматко был очень оригинален, но в то же время не всегда справедлив. И вот однажды в роту из штаба дивизии прибыла специальная комиссия для проверки справедливости способа выбора. Комиссию интересовало, какие солдаты чаще всего и какие реже всего были в наряде за последние  $K$  дней.

Помню, был у нас тогда в роте программист один, Пупкин Вася. Вот мы ему программу для комиссии сделать и поручили. Он со своей работой справился отлично. А вы сможете?»

Вам необходимо определить максимальное количество нарядов, выпавшее одному солдату в течение  $K$  дней (кому-то ведь не повезло больше всех), а также количество солдат, получивших наряд максимальное количество раз. Вам также нужно подсчитать аналогичные два числа для минимального случая.

### Входные данные

Во входном файле содержатся два целых числа:  $N$  ( $1 \leq N \leq 1000000$ ) – количество солдат в списке и  $K$  ( $1 \leq K < 2^{31}$ ) – количество дней в рассматриваемом комиссией периоде.

### Выходные данные

В выходной файл выведите два строки, содержащие по два целых числа каждая. Первая строка соответствует случаю максимального количества нарядов, а вторая – минимальному. Первое число в каждой строке должно выражать соответствующее количество нарядов, а второе – количество солдат, получивших такое количество нарядов.

### Пример

| <code>army.in</code> | <code>army.out</code> | Примечание  |
|----------------------|-----------------------|---|
| 3 2                  | 2 1<br>1 2            | За два дня третий солдат получил два наряда, а оставшиеся два солдата – по одному наряду.   |
| 6 2                  | 2 2<br>0 2            | За два дня солдаты под номерами 3 и 5 получили по два наряда, солдаты под номерами 2 и 4 – по одному, а солдаты 1 и 6 вообще бездельничали. |

## Задача К. Расстановка офицеров

Имя входного файла: `officers.in`  
Имя выходного файла: `officers.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 Мб

Ординарец Тамары Семеновны Иванов-оглы-Писемский очень любил вспоминать случаи из армейской жизни. Однажды, когда Шарик с Печкиным на рыбалку пошли, а Матроскин в огороде работал, Иванов-оглы рассказал дяде Федору и его папе, которые как раз читали книжку по программированию, интересную историю.

«Однажды должны были в нашей части учения военные по информатике пройти. Начальства разного большого понаехало. Даже сам генерал-лейтенант-программист Борис Борисович Бинар приехал. А к учением нужно подготовиться. Тем более генерал-программисты народ странный и требования у них странные.

Для учений нужно определить ранг офицеров, которые будут находиться в каждом из  $N$  командных пунктов. Командные пункты пронумерованы числами от 1 до  $N$ , которые соответствуют расстояниям от командных пунктов до главного военно-командного центра (ГВКЦ). В командных пунктах могут находиться только офицеры двух рангов – 0 и 1, причем в каждом командном пункте могут находиться только офицеры одного ранга.

Более того, Бинар изготовил план учений из  $M$  положений,  $i$ -ое из которых требует, чтобы количество пунктов с офицерами ранга 1 среди пунктов с номерами с  $A_i$  по  $B_i$  включительно было не меньше  $C_i$ .

Другая бы товарищ полковник в такой ситуации растерялась», – продолжал Иванов-оглы, – «но наша товарищ полковник не такая. Она даже выставила дополнительное требование к расстановке.

Пронумеруем пункты с офицерами с рангом 0 от 1 до  $P$  в порядке возрастания расстояния от ГВКЦ, где  $P$  – общее количество таких пунктов. Тогда в выходной расстановке пункт 1 должен быть как можно ближе к ГВКЦ. Если таких расстановок несколько, то товарищ полковник хотела минимизировать расстояние от пункта 2 до ГВКЦ и т.д. Вот такая сложная задача, но Тамара Семеновна сделала ее в два счета. А вы сможете повторить ее подвиг?»

### Входные данные

Первая строка входного файла содержит два натуральных числа  $N$  и  $M$  ( $1 \leq N \leq 100000$ ,  $1 \leq M \leq 100000$ ). В следующих  $M$  строках идет описание положений плана Бинара. В  $(i+1)$ -ой строке содержится 3 натуральных числа  $A_i$ ,  $B_i$ ,  $C_i$  ( $1 \leq A_i \leq B_i \leq N$ ,  $0 \leq C_i \leq B_i - A_i + 1$ ).

### Выходные данные

Если расстановка невозможна, то в единственную строку выходного файла выведите строку “No possible arrangement.” (без кавычек). Иначе выведите единственную строку из  $N$  цифр 0 или 1,  $i$ -ый символ которого соответствует рангу офицеров в  $i$ -ом пункте.

### Пример

| <code>officers.in</code>       | <code>officers.out</code> |
|--------------------------------|---------------------------|
| 5 3<br>1 2 1<br>2 4 2<br>3 5 2 | 01011                     |