

## Задача А. Игра на графе

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Геннадий и Георгий играют в интересную игру на ориентированном графе. В графе  $n$  вершин и  $m$  рёбер, возможны петли. Перед началом игры Геннадий и Георгий поставили в какую-то вершину фишку. Игроки ходят по очереди, перемещая фишку вдоль какого-то ребра, исходящего из вершины, в которой она находится. Если у игрока нет ни одного хода, он проигрывает.

Ваша задача — для каждой вершины графа и для обоих игроков определить результат игры, если фишка стоит в этой вершине и если этот игрок ходит первым. Звучит просто? Всё не совсем так.

С одной стороны, Геннадий получает удовольствие от процесса игры, и хочет, чтобы игра длилась как можно дольше. Он даже предпочтёт бесконечную игру победе. Но если он не может играть бесконечно, он, очевидно, предпочтёт поражению победу.

С другой стороны, у Георгия куча дел, и он не хочет играть бесконечно. Георгий хочет победить, но если он не может, он предпочтёт поражение бесконечной игре.

Оба игрока играют оптимально. Оба игрока знают предпочтения противника.

### Формат входных данных

В первой строке записано два целых числа — число вершин  $n$  ( $1 \leq n \leq 100\,000$ ) и число рёбер  $m$  ( $1 \leq m \leq 200\,000$ ). В следующих  $m$  строках записаны по два числа  $a$  и  $b$ , обозначающих ребро из вершины  $a$  в вершину  $b$ . Вершины пронумерованы от 1 до  $n$ . Каждая пара  $(a, b)$  встречается не более одного раза.

### Формат выходных данных

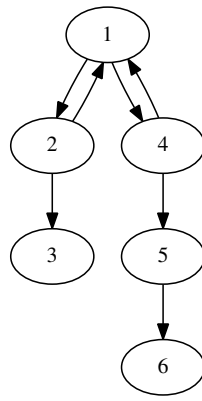
В первой строке выведите  $n$  символов —  $i$ -й символ означает результат игры, если Геннадий начинает в вершине  $i$ . Во второй строке выведите  $n$  символов —  $i$ -й символ означает результат игры, если Георгий начинает в вершине  $i$ . Результат игры обозначается буквой «W», если начинающий игру игрок побеждает, «L» — если проигрывает, и «D», если игра продолжается бесконечно.

### Пример

стандартный ввод	стандартный вывод
6 7	WDLDWL
1 2	DWLLWL
2 1	
2 3	
1 4	
4 1	
4 5	
5 6	

### Замечание

Вершины 3 и 6 изначально проигрывают. В вершине 5 единственный ход ведёт в вершину 6, и игрок выигрывает. Если Георгий начинает в вершине 1, или если Геннадий начинает в вершинах 2 или 4, Геннадий может всегда пойти в вершину 1 и сделать игру бесконечной. Если Георгий начинает в вершине 4, он может пойти или в вершину 1 (что приведёт к ничьей) или в вершину 5, что приведёт к поражению. Георгий предпочитает проиграть. Аналогично, из вершины 2, он предпочитает пойти в вершину 3 и победить. Из вершины 1, Геннадий может пойти в вершину 2 и проиграть, или в вершину 4 и победить. Он предпочитает победить.



## Задача В. Малыш и Карлсон

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

На свой День рождения Малыш позвал своего лучшего друга Карлсона. Мама испекла его любимый пирог прямоугольной формы  $a \times b \times c$  сантиметров. Карлсон знает, что у Малыша еще есть килограмм колбасы. Чтобы заполучить ее, он предложил поиграть следующим образом: они по очереди разрезают пирог на две ненулевые по объему прямоугольные части с целыми измерениями и съедают меньшую часть (в случае, когда части равные, можно съесть любую). Проигрывает тот, кто не может сделать хода (то есть когда размеры будут  $1 \times 1 \times 1$ ). Естественно, победителю достается колбаса.

Малыш настаивает на том, чтобы он ходил вторым.

Помогите Карлсону выяснить, сможет ли он выиграть, и если сможет — какой должен быть его первый ход для этого.

Считается, что Малыш всегда ходит оптимально.

### Формат входных данных

Во входном файле содержится 3 целых числа  $a, b, c$  ( $1 \leq a, b, c \leq 5000$ ) — размеры пирога.

### Формат выходных данных

В случае, если Карлсон не сможет выиграть в Малыша, выведите NO. В противном случае в первой строке выведите YES, во второй — размеры пирога после первого хода Карлсона в том же порядке, что и во входном файле.

### Примеры

стандартный ввод	стандартный вывод
1 1 1	NO
1 2 1	YES 1 1 1
1 1 10	YES 1 1 7

## Задача С. Королевская игра

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Император Шардонии и король Флатляндии во время своей встречи решили сыграть в новую настольную игру. Разумеется, играть они будут шариками на клетчатом плоском поле.

Поле для игры представляет из себя клетчатый прямоугольник, строки которого пронумерованы от 0 до 100, а столбцы — от 0 до 100. По некоторым клеткам поля будут раскиданы  $N$  шариков. Игроки ходят по очереди. В свой ход игрок может взять шарик с позиции  $(l_i, c_i)$  и выбрать положительное целое число  $u$ . После этого игрок может передвинуть этот шарик в одну из следующих ячеек:

- $(l_i - u, c_i)$
- $(l_i, c_i - u)$
- $(l_i - u, c_i - u)$

Разумеется, ход можно сделать только если соответствующая клетка существует на поле. Выигрывает игрок, который смог передвинуть какой-либо шарик в позицию  $(0, 0)$ .

Вы — главный советник императора Шардонии. Он вызвался ходить первым, и спрашивает вас, может ли он выиграть, если оба игрока будут играть оптимально. Ответьте ему на вопрос, потому что иначе вас казнят.

### Формат входных данных

На вход программе подается число  $N$  ( $1 \leq N \leq 1000$ ) — количество шариков на поле.

В следующих  $N$  строках расположены числа  $l_i, c_i$  ( $0 \leq l_i, c_i \leq 100$ ) — координаты шариков.

### Формат выходных данных

Вы должны вывести символ «Y», если Император может выиграть, и «N» иначе.

### Примеры

стандартный ввод	стандартный вывод
2 1 3 2 3	Y
1 1 2	N

## Задача D. Кучки с конфетами

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Есть  $n$  кучек с конфетами на столе. Они пронумерованы целыми числами от 1 до  $n$ , в  $i$ -й кучке находится  $a_i$  конфет.

Два игрока играют в игру. За ход можно сделать одно из двух действий:

1. Выбрать любую кучку, в которой осталось максимальное число конфет и съесть все конфеты из этой кучки;
2. Съесть по 1 конфете из всех непустых кучек.

Игрок, который съест последнюю конфету на столе проигрывает. Определите, кто выигрывает при правильной игре.

### Формат входных данных

В первой строке находится единственное целое число  $n$  ( $1 \leq n \leq 10^5$ ) — количество кучек. Во второй строке находится  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — количества конфет в каждой из кучек.

### Формат выходных данных

Если первый игрок имеет выигрышную стратегию, выведите «First» (без кавычек), иначе выведите «Second» (без кавычек).

### Примеры

стандартный ввод	стандартный вывод
2 1 3	First
3 1 2 1	First
3 1 2 3	Second

## Задача Е. Кто выше?

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

Костя с Ваней, не сговариваясь, подарили друг другу на Новый год по дереву. Разумеется, в новогоднюю ночь обязательно надо сыграть в какую-нибудь игру. Поскольку у наших героев при себе были только подаренные деревья, то играть они решили каждый со своим деревом.

Игра с деревом выглядела следующим образом — игрок должен выбрать вершину в своем дереве. После этого дерево подвешивается за соответствующую вершину. Количество очков, которое набрал игрок, определяется как максимальная глубина вершины в корневом дереве. Глубиной вершины  $v$  в дереве с корнем  $r$  считается количество вершин на простом пути от  $r$  до  $v$ .

Костя с Ваней договорились ровно в полночь выбрать вершину, а потом сравнить количество набранных очков в игре. Тот, кто набирает больше очков, побеждает. Если набрано одинаковое количество очков, то не побеждает никто. Подведение итогов договорились провести на следующий день.

Костя настолько увлекся празднованием, что абсолютно забыл, какую же вершину он выбрал. Единственное, что Костя запомнил — Ваня подошел к состязанию серьезно и выбрал вершину так, чтобы максимизировать количество своих очков.

Утром следующего дня Костю заинтересовал вопрос: «А может ли так случиться, что я точно победил, вне зависимости от выбранной вершины?». Помогите Косте ответить на этот вопрос. Учтите, что он мог выбрать абсолютно любую вершину, в отличие от Вани.

### Формат входных данных

В первой строке вводится число  $N$  ( $1 \leq N \leq 5 \cdot 10^5$ ) — количество вершин в дереве Кости.

Следующие  $N - 1$  строк содержат по 2 числа:  $u, v$  ( $1 \leq u, v \leq N, v \neq u$ ) — что означает, что вершины  $u$  и  $v$  соединены ребром в дереве Кости.

В первой строке вводится число  $M$  ( $1 \leq M \leq 5 \cdot 10^5$ ) — количество вершин в дереве Вани.

Следующие  $M - 1$  строк содержат по 2 числа:  $u, v$  ( $1 \leq u, v \leq M, v \neq u$ ) — что означает, что вершины  $u$  и  $v$  соединены ребром в дереве Вани.

### Формат выходных данных

Выведите «GGEZ», если Костя выиграл независимо от выбора вершины. Иначе выведите «FF».

## Примеры

стандартный ввод	стандартный вывод
7 1 2 2 5 3 6 2 4 1 3 3 7 5 1 2 1 3 2 4 2 5	FF
7 1 2 1 3 3 4 4 5 2 6 6 7 5 1 2 1 3 2 4 2 5	FF

## Задача F. Смак

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Правильно ли я понимаю, что наше рабочее место на данной кулинарной передаче ограничено одним столом, одним тестом и тремя мисками? Вы же понимаете, что мы просто физически здесь не поместимся?

Михаил Анопренко

Иван Ургант пригласил Ивана Сафонова на свою кулинарную передачу «Смак». В сегодняшнем выпуске они будут готовить огромный пирог из теста, которое уже заготовлено помощниками Урганта и налито в миску. В миске всегда находится целое неотрицательное число граммов теста. Оба Ивана, начиная с Урганта, по очереди применяют один из двух кулинарных приёмов:

1. Набрать мерным стаканом один грамм теста из миски и добавить в форму для пирога.
2. Взбить тесто миксером. При этом часть теста расплёскивается. Если в миске перед взбиванием было  $G$  граммов теста, то останется  $\varphi(G)$  граммов теста.  $\varphi$  — это функция Эйлера, которая по данному положительному целому числу  $G$  возвращает количество целых чисел от 1 до  $G$  включительно, которые взаимно просты с  $G$ , то есть не имеют с  $G$  общих делителей, больших единицы. Приведём таблицу, в которой вычислены некоторые значения этой функции:

$G$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$\varphi(G)$	1	1	2	2	4	2	6	4	6	4	10	4	12	6	8	8	16	6	18	8

Когда тесто в миске полностью закончится, пирог будет считаться приготовленным; более того, тот из Иванов, чей приём потратил последний грамм теста в миске, и будет считаться поваром, приготовившим пирог. Считая, что оба Ивана желают считаться создателями пирога и будут для этого каждый раз оптимально выбирать один из двух приёмов, определите, чей кулинарный приём завершит готовку.

### Формат входных данных

В первой строке находится одно целое положительное число  $G$  ( $1 \leq G \leq 10^9$ ) — изначальное количество теста в граммах.

### Формат выходных данных

Если Иван Ургант может гарантированно завершить готовку пирога, в первой строке выведите «Ivan Urgant». Во второй строке выведите «Draw», если для этого Урганту нужно первым действием набрать грамм теста из миски и налить в форму, или же выведите «Mix», если первым действием ему надо взбить тесто. Если Ургант при выполнении любого из двух приёмов может гарантировать, что именно он завершит готовку пирога, то вы можете вывести любое одно из двух слов «Draw» и «Mix».

Если Иван Сафонов может гарантированно завершить готовку пирога, в первой строке выведите «Ivan Safonov».

## Примеры

стандартный ввод	стандартный вывод
1	Ivan Urgant Draw
2	Ivan Safonov
4	Ivan Urgant Mix
14	Ivan Urgant Draw

## Замечание

Если в миске один грамм теста, Ургант просто должен налить его в форму и сразу завершить готовку пирога.

Если в миске два грамма теста, то любой из двух кулинарных приёмов Урганта приведёт к тому, что в миске останется один грамм теста, после чего Сафонов легко доготовит пирог.

Если в миске четыре грамма теста, то Ургант проиграет, если нальёт один грамм теста в форму, и выиграет, если вместо этого он взобьёт тесто и оставит в миске  $\varphi(4) = 2$  грамма.

Если в миске 14 граммов теста, то всё наоборот: если Иван Ургант его взобьёт, то оставит в миске  $\varphi(14) = 6$  граммов теста, в результате чего он проиграет; если же он нальёт один грамм теста в форму, то в миске останется 13 граммов теста, и можно доказать, что в такой ситуации Иван Сафонов не может гарантировать, что именно он приготовит пирог.

## Задача G. Дровосек

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Двое играют в следующую игру: имеется дерево с отмеченной вершиной (корнем). Игроки ходят по очереди. За ход игрок разрубает ветку (стирает ребро), причем из двух получившихся компонент связности остается только та, которая содержит корень — остальная отваливается и больше в игре не участвует. Проигрывает тот, кто не может сделать ход.

Определите, может ли выиграть первый игрок, и если да, то укажите любой из его выигрышных ходов.

### Формат входных данных

В первой строке входного файла находится 2 числа  $n$  и  $r$  — количество вершин дерева и номер корня ( $2 \leq n \leq 100\,000$ ,  $1 \leq r \leq n$ ). Далее следует  $n - 1$  строк, в каждой из которых находятся два числа — номера вершин, которые соединяет очередное ребро.

### Формат выходных данных

Выведите в выходной файл одно число: 1 или 2 — номер игрока, который выигрывает при правильной игре. Если выигрывает первый игрок, то выведите также любой его выигрышный ход, т.е. порядковый номер ребра во входном файле, которое ему достаточно разрубить первым ходом (число от 1 до  $n - 1$ ).

### Примеры

стандартный ввод	стандартный вывод
5 5 2 3 1 3 2 5 4 5	1
1 1	2

## Задача N. Игра с клавиатурой

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Маша и Миша играют в интересную игру. Каждый из них выписывает набор слов, на чём первая стадия игры заканчивается. Во второй стадии игры участники берут клавиатуру и по очереди выламывают из неё клавиши, соответствующие латинским буквам, всего  $l$  штук. После этого игроки считают количество своих слов, которые можно напечатать с помощью оставшихся клавиш. После этого тот, чьих слов остаётся меньше, проигрывает сопернику количество конфет, равное разности количеств оставшихся слов.

Первая стадия игры уже завершена, первый ход по жребию (или по мишиной галантности) предстоит Маше. Определите, сколько конфет она может себе гарантировать при оптимальной игре обоих.

### Формат входных данных

В первой строке входного файла записано число  $l$  — количество клавиш, которые будут выломаны за всю игру ( $1 \leq l \leq 26$ ). Далее записаны наборы машинных и мишинных слов в следующем формате: на одной строке количество слов, на следующей — сами слова, разделённые пробелами.

Все слова состоят из строчных букв латинского алфавита. Каждый игрок выписал не более 15 непустых слов, состоящих из не более, чем 30 букв каждое.

### Формат выходных данных

Выведите единственное целое число — выигрыш Маши при оптимальной игре. Если Маша вынуждена проиграть, выведите её минимальный проигрыш со знаком «минус».

### Пример

стандартный ввод	стандартный вывод
2	1
5	
abacaba a zxxzyz trava abc	
1	
a	

## Задача I. Чапаев на дереве

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	512 мегабайт

Вова и Марина любят играть в игры, а особенно — придумывать к ним свои правила. Недавно они открыли для себя веселую игру «Чапаев», в которой игроки должны сбивать щелчками шашки вражеского цвета с шахматной доски (также эта игра известна под названием «Щелкунчики»). Вдоволь наигравшись, они решили модифицировать правила, добавив игре математическую сложность.

Теперь они играют в «Чапаева» не на шахматной доске, а на доске в форме дерева! Их дерево состоит из  $N$  вершин. Вершина 1 является корнем дерева, а из каждой из оставшихся вершин проведено ребро в некоторую вершину с меньшим номером — её непосредственного предка.

В игре участвуют шашки одного цвета, изначально расположенные в некоторых вершинах дерева. За один ход игрок выбирает некоторую шашку и щелчком отправляет ее к корню по ребрам дерева, сбивая при этом с доски все встреченные на пути шашки. Сама шашка, по которой производился удар, после попадания в корень дерева также слетает с доски.

Игроки делают ходы по очереди. Проигрывает тот игрок, к ходу которого на доске не остается шашек.

Придуманная ими игра замечательна также тем, что на одной и той же доске можно играть, начиная с разных начальных позиций шашек. Практика показала, что самые интересные партии получаются, если исходно расставить фишки во все вершины, являющиеся потомками (непосредственными или косвенными) некоторой вершины  $Root$ , при этом в саму вершину  $Root$  фишка не ставится.

Дети решили сыграть  $N$  партий, перебрав в качестве вершины  $Root$  каждую вершину дерева по одному разу. Если у очередной вершины  $Root$  нет потомков, и на доске исходно не оказывается ни одной фишки, то игры не происходит, и дети переходят к следующей расстановке. В каждой партии Марина ходит первой.

Вова интересуется у вас, в скольких партиях Марина сможет одержать победу, если игроки будут действовать оптимально.

### Формат входных данных

В первой строке находится целое число  $N$  ( $1 \leq N \leq 500\,000$ ) — количество вершин в дереве.

Во второй строке следуют целые числа  $p_2, p_3, \dots, p_N$ , разделенные пробелами, где  $p_i$  — это номер вершины, являющейся предком вершины  $i$  ( $1 \leq p_i \leq i$ ).

### Формат выходных данных

Выведите единственное целое число — количество партий, в которых Марина одержит победу.

### Пример

стандартный ввод	стандартный вывод
7 1 2 3 1 5 5	3

### Замечание

Разберем тест из условия. Доска для игры показана на рисунках ниже. Дети сыграют четыре партии, выбирая в качестве  $Root$  вершины 1, 2, 3 и 5. Если выбрать в качестве  $Root$  любую из трех оставшихся вершин, на доске исходно не окажется ни одной фишки, поэтому игры не произойдет.

Если выбрать в качестве  $Root$  вершину 5, фишки будут исходно находиться в вершинах 6 и 7. В такой партии Марина проигрывает: после того, как она сбивает любую из этих двух фишек с доски, Вова сбивает оставшуюся и заканчивает партию.

Если выбрать в качестве  $Root$  вершину 2 или 3, у Марины будет возможность выиграть игру за один ход, щелкнув по фишке из вершины 4 (при этом, в случае  $Root = 2$ , она по пути также собьет фишку из 3 вершины по правилам игры)

Можно убедиться, что если выбрать в качестве *Root* вершину 1, у Марины также будет выигрышная стратегия. Для этого первым ходом Марина должна сбить фишку из вершины 2. Пример партии с таким начальным расположением показан ниже.

Таким образом, Марина выигрывает в трех партиях

## Задача J. Альфа Дерево

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

У вас есть полное бинарное дерево глубины  $n$  ( $0 \leq n \leq 30$ ).

В дереве  $2^n$  листьев, они пронумерованы слева направо числами от 0 до  $2^n - 1$ .

В  $i$ -м листе записано число  $x_i = (ai^2 + bi + c) \bmod m$ .

Есть фишка, которая изначально находится в корне дерева. Двое играют в игру, двигая фишку вниз по дереву. Когда фишка достигает листа дерева, игра заканчивается. Цель первого игрока — максимизировать число в листе, цель второго — минимизировать.

### Формат входных данных

Числа  $n, a, b, c, m$ . При этом  $10 \leq m \leq 10^9$ .

Все  $a, b, c$  сгенерированы равномерным распределением на  $[0, m)$ .

### Формат выходных данных

Выведите результат игры при оптимальной игре обоих.

### Система оценки

Оценка потестовая. Все тесты независимы.

### Пример

стандартный ввод	стандартный вывод
3 10 7 9 20	11

### Замечание

Взятие остатка по модулю — небыстрая операция. Чем их меньше, тем лучше.

## Задача К. Самый честный Ним

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Алиса и Боб играют в игру с  $n$  кучками камней. Гарантируется, что  $n$  — четное. В  $i$ -й кучке  $a_i$  камней.

Алиса и Боб ходят по очереди, Алиса ходит первой.

На каждом ходу игрок должен выбрать **ровно**  $\frac{n}{2}$  непустых куч и независимо удалить положительное количество камней из каждой из выбранных куч. Игрок **может** удалить **разное** количество камней из каждой из них за один ход. Первый игрок, который не может сделать ход, проигрывает (когда меньше, чем  $\frac{n}{2}$  непустых куч).

Дана начальная конфигурация, по ней определите, кто победит в игре.

### Формат входных данных

В данной задаче несколько наборов входных данных. В первой строке вводится количество наборов входных данных  $t$  ( $1 \leq t \leq 10^4$ ).

В первой строке каждого набора входных данных содержится одно целое число  $n$  ( $2 \leq n \leq 50$ ) — количество кучек. Гарантируется, что  $n$  — четное.

Во второй строке каждого набора входных данных содержится  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 50$ ) — количество камней в каждой кучке.

### Формат выходных данных

Выведите «Alice», если победит Алиса, если же победит Боб, то выведите «Bob» (без кавычек).

### Пример

стандартный ввод	стандартный вывод
3	Bob
2	Alice
8 8	Alice
4	
1 2 3 3	
6	
1 2 3 4 5 6	

### Замечание

В первом наборе входных данных каждый игрок может удалить камни только из одной кучи ( $\frac{2}{2} = 1$ ). Алиса проигрывает, так как Боб может копировать действия Алисы, поэтому у Алисы первой закончатся ходы.

Во втором примере Алиса может убрать 2 камня из первой кучки и 3 камня из третьей кучки за первый ход, чтобы гарантировать победу.