

## Содержание

<b>Халявный АС.</b>	<b>2</b>
1 Задача А. Калькулятор [1 секунда, 64 mb]	2
<b>Простые игры</b>	<b>3</b>
2 Задача В. Одна кучка [2 секунды, 64 mb]	3
3 Задача С. Вариация Нима [2 секунды, 256 mb]	4
4 Задача D. Комфортабельная рассадка [2 секунды, 256 mb]	5
<b>Задачи</b>	<b>6</b>
5 Задача Е. Малыш и Карлсон [1 секунда, 64 mb]	6
6 Задача F. Ретроанализ для маленьких [1 секунда, 256 mb]	7
7 Задача G. Woodcut. Дровосек [2 секунды, 256 mb]	8

## Халявный АС.

### 1 Задача А. Калькулятор [1 секунда, 64 mb]

Имеется калькулятор, который выполняет следующие операции:

- Умножить число  $X$  на 2.
- Умножить число  $X$  на 3.
- Прибавить к числу  $X$  единицу.

Определите, какое наименьшее количество операций требуется, чтобы получить из числа 1 число  $N$ .

#### Формат входных данных

Во входном файле написано натуральное число  $N$ , не превосходящее  $10^6$ .

#### Формат выходных данных

В первой строке выходного файла выведите минимальное количество операций. Во второй строке выведите числа, последовательно получающиеся при выполнении операций. Первое из них должно быть равно 1, а последнее  $N$ .

#### Примеры

стандартный ввод	стандартный вывод
1	0 1
5	3 1 3 4 5
962340	17 1 3 9 27 54 55 165 495 1485 4455 8910 17820 17821 53463 160389 160390 481170 962340

## Простые игры

### 2 Задача В. Одна кучка [2 секунды, 64 mb]

Два игрока играют в игру. На столе лежит кучка из  $N$  камней. Двое ходят по очереди. За ход можно взять  $a_1, a_2, \dots, a_k$  камней. Проигрывает тот, кто не может сделать ход. Определите победителя!

#### Формат входных данных

В первой строке записано число  $k$ . Во второй строке  $k$  чисел —  $a_1, a_2, \dots, a_k$ . В третьей строке идет число  $m$  — количество различных  $N$ , для каждого из которых требуется определить победителя. В четвертой строке  $m$  чисел —  $N_1, N_2, \dots, N_m$ .

Ограничения:  $1 \leq k \leq 20, m \leq 10^4, 1 \leq N_i, a_i \leq 10^6$ .

#### Формат выходных данных

Выведите  $m$  строк, в каждой ответ на вопрос “кто выиграет” — `First` или `Second`.

#### Пример

стандартный ввод	стандартный вывод
3	First
1 2 3	First
8	First
1 2 3 4 5 6 7 8	Second
	First
	First
	First
	Second

### 3    Задача С. Вариация Нима [2 секунды, 256 mb]

На столе лежат  $n$  кучек камней:  $a_1$  камней в первой кучке,  $a_2$  камней во второй,  $\dots$ ,  $a_n$  в  $n$ -ой. Двое играют в игру, делая ходы по очереди. За один ход игрок может либо взять произвольное ненулевое количество камней (возможно, все) из одной любой кучки, либо произвольным образом разделить любую существующую кучку, в которой не меньше двух камней, на две непустые кучки. Проигрывает тот, кто не может сделать ход. Кто выигрывает при правильной игре?

#### Формат входных данных

В первой строке задано целое число  $t$  — количество тестов ( $1 \leq t \leq 100$ ). Следующие  $t$  строк содержат сами тесты. Каждая из них начинается с целого числа  $n$  — количества кучек ( $1 \leq n \leq 100$ ). Далее следует  $n$  целых чисел  $a_1, a_2, \dots, a_n$  через пробел — количество камней в кучках ( $1 \leq a_i \leq 10^9$ ).

#### Формат выходных данных

Выведите  $t$  строк; в  $i$ -ой строке выведите “FIRST”, если в  $i$ -ом тесте при правильной игре выигрывает первый игрок, и “SECOND”, если второй.

#### Пример

стандартный ввод	стандартный вывод
3	FIRST
1 1	SECOND
2 1 1	FIRST
3 1 2 3	

#### 4 Задача D. Комфортабельная рассадка [2 секунды, 256 mb]

В парке стоит скамейка на  $n$  мест. Изначально она пуста. Две команды играют в игру, по очереди *комфортабельно* сажая на скамейку по одному человеку. Человек садится *комфортабельно*, если на соседних местах ни справа, ни слева от него непосредственно никого нет. Проигрывает та команда, которая не может *комфортабельно* посадить очередного человека на своём ходу.

Дано  $n$  — количество мест на скамейке. Какая команда выигрывает при правильной игре?

##### Формат входных данных

В первой строке входного файла задано целое число  $n$  ( $1 \leq n \leq 100$ ).

##### Формат выходных данных

В первой строке выходного файла выведите “FIRST”, если при правильной игре выигрывает первая команда, и “SECOND”, если вторая.

##### Примеры

стандартный ввод	стандартный вывод
2	FIRST
4	SECOND

## Задачи

### 5 Задача Е. Малыш и Карлсон [1 секунда, 64 mb]

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  $a, b, c$  ( $1 \leq a, b, c \leq 5\,000$ ) — 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
2 1 1	YES 1 1 1

## 6 Задача F. Ретроанализ для маленьких [1 секунда, 256 mb]

Дан ориентированный весёлый граф из  $n$  вершин и  $m$  ребер. Оля и Коля в игру. Изначально фишка стоит в вершине  $i$ . За ход можно передвинуть фишку по любому из исходящих ребер. Тот, кто не может сделать ход, проигрывает. Ваша задача — для каждой вершины  $i$  определить, кто выиграет при оптимальной игре обоих.

### Формат входных данных

Входные данные состоят из одного или нескольких тестов. Каждый тест содержит описание весёлого ориентированного графа. Граф описывается так: на первой строке два целых числа  $n$  ( $1 \leq n \leq 300\,000$ ) и  $m$  ( $1 \leq m \leq 300\,000$ ). Следующие  $m$  строк содержат ребра графа, каждое описывается парой целых чисел от 1 до  $n$ . Пара  $a\ b$  обозначает, что ребро ведет из вершины  $a$  в вершину  $b$ . В графе могут быть петли, могут быть кратные ребра. Сумма  $n$  по всем тестам не превосходит 300 000, сумма  $m$  по всем тестам также не превосходит 300 000.

### Формат выходных данных

Для каждого теста выведите для каждой вершины `FIRST`, `SECOND` или `DRAW` в зависимости от того, кто выиграет при оптимальной игре из этой вершины. Ответы к тестам разделяйте пустой строкой.

### Примеры

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

## 7 Задача G. Woodcut. Дровосек [2 секунды, 256 mb]

Двое играют в следующую игру: имеется дерево с отмеченной вершиной (корнем). Игроки ходят по очереди. За ход игрок рубит 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 находятся 2 числа,  $N$  и  $R$  — количество вершин дерева и номер корня ( $1 < N \leq 100\,000$ ,  $1 \leq R \leq N$ ). Далее следуют  $N - 1$  строка, 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 разорвать первым ходом (число от 1 до  $N - 1$ ).

### Пример

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