



## II (муниципальный) этап Всероссийской олимпиады школьников 2011/2012 уч. год. Новосибирская область

Предмет	Класс	Дата	Время начала	Время окончания
<b>Информатика</b>	<b>7-11</b>	<b>24.11.2011</b>	<b>10:00</b>	<b>15:00</b>

### Для всех задач:

Имя входного файла:	<i>input.txt</i>
Имя выходного файла:	<i>output.txt</i>
Ограничение по времени:	1 секунда на тест
Ограничение по памяти:	256 Мб

### Задача 1. На первый-второй

На уроке физкультуры классу нужно было разделиться на  $k$  команд. Чтобы не было споров, учитель отдал команду рассчитаться на первый –  $k$ -й. Тот, кто при расчете оказался под номером  $i$ , должен идти в команду с номером  $i$ . В результате таких перестроений в некоторых командах участников получилось меньше, чем в остальных.

Требуется определить наименьший номер команды с меньшим количеством участников.

#### Входные данные

В первой строке входного файла записано два целых числа  $N$  и  $k$  — количество учеников в классе и количество команд ( $0 < N, k \leq 10^9$ ).

#### Выходные данные

В выходной файл необходимо вывести наименьший номер команды с меньшим количеством участников. Если во всех командах одинаковое количество участников, то нужно вывести число 0.

#### Примеры

<i>input.txt</i>	<i>output.txt</i>
10 3	2
10 2	0

### Задача 2. Обработка статистических данных

Британские учёные провели серию из  $N$  экспериментов и получили набор значений, которые являются целыми положительными числами  $a_1 \dots a_N$ . Они хотят найти среднее арифметическое этих чисел. Также их интересует, какие из значений  $a_i$  содержатся в наборе наиболее часто. От вас требуется написать программу, которая поможет британским учёным — определит среднее значение и найдет наиболее часто встречающиеся числа.

#### Входные данные

На первой строке входного файла записано одно целое число  $N$  — количество экспериментов ( $1 \leq N \leq 1000$ ).

Во второй строке записано через пробел  $N$  целых чисел  $a_1 \dots a_N$  — значения величин, каждое из которых получено в результате соответствующего по порядку проведения эксперимента. Все числа по модулю не превосходят 100.

#### Выходные данные

В первую строку выходного файла необходимо выдать одно целое число — среднее значение всех заданных величин с округлением до ближайшего целого.

Во второй строке должно быть записано целое число  $K$  — количество различных наиболее часто встречающихся значений, а в третью строку необходимо записать через пробел эти  $K$  наиболее часто встречающихся целых чисел, упорядоченные по возрастанию.

#### Примеры

<i>input.txt</i>	<i>output.txt</i>
10 3 1 3 2 3 4 1 2 1 4	2 2 1 3
5 1 3 5 2 4	3 5 1 2 3 4 5



## II (муниципальный) этап Всероссийской олимпиады школьников 2011/2012 уч. год. Новосибирская область

Предмет	Класс	Дата	Время начала	Время окончания
<b>Информатика</b>	<b>7-11</b>	<b>24.11.2011</b>	<b>10:00</b>	<b>15:00</b>

### Задача 3. Новая игра

Вы пишете новую двумерную игру *AngryNerds*. Суть игры состоит в том, чтобы попасть летящим по параболе снарядом в неподвижный объект. Неподвижный объект — это прямоугольник, который задается неравенствами  $x_0 \leq x \leq x_1$ ,  $y_0 \leq y \leq y_1$ . Предмет начинает полёт в точке с координатой  $x = 0$  и летит в направлении возрастания  $x$ . Параметры параболы определяются действиями пользователя, а ваша текущая задача состоит в том, чтобы по заданным коэффициентам  $a$ ,  $b$  и  $c$  траектории полёта  $y = ax^2 + bx + c$  определить, попадёт ли предмет в цель.

#### Входные данные

В первой строке входного файла записано через пробел четыре целых числа  $x_0$ ,  $y_0$ ,  $x_1$ ,  $y_1$ , которые задают координаты левого нижнего и правого верхнего углов неподвижного прямоугольника, куда нужно попасть ( $0 < x_0 < x_1 \leq 100$ ,  $0 < y_0 < y_1 \leq 100$ ).

В следующей строке задается целое число  $N$  — количество попыток пользователя поразить цель ( $1 \leq N \leq 10$ ).

Далее в  $N$  строках описываются попытки пользователя, по одной в строке. Каждая попытка — это три целых числа, записанные через пробел  $a$ ,  $b$  и  $c$  — коэффициенты параболы, по которой происходит выстрел ( $-100 \leq a < 0$ ,  $-100 \leq b$ ,  $c \leq 100$ ).

#### Выходные данные

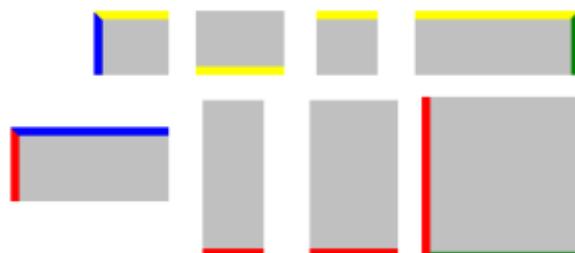
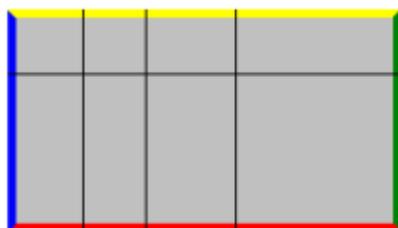
Для каждой попытки пользователя в выходной файл по одному на строке в порядке перечисления попыток во входном файле необходимо выдать слово **YES**, если предмет пролетает через целевой прямоугольник или задевает его и **NO**, в противном случае.

#### Примеры

<i>input.txt</i>	<i>output.txt</i>
<pre>3 4 6 7 4 -1 2 0 -1 8 0 -1 4 0 -1 6 0</pre>	<pre>NO NO NO YES</pre>
<pre>1 8 2 9 3 -2 12 -10 -3 18 -15 -2 12 -6</pre>	<pre>NO YES YES</pre>

### Задача 4. Прямоугольник

Через прямоугольник, все четыре стороны которого раскрашены в разные цвета, провели несколько разрезов, параллельных его сторонам. После этого получившиеся маленькие прямоугольники перемешали и, возможно, несколько раз повернули на 90 градусов по часовой стрелке. На рисунке приведен пример разрезания прямоугольника на маленькие части.





## II (муниципальный) этап Всероссийской олимпиады школьников 2011/2012 уч. год. Новосибирская область

Предмет	Класс	Дата	Время начала	Время окончания
<b>Информатика</b>	<b>7-11</b>	<b>24.11.2011</b>	<b>10:00</b>	<b>15:00</b>

Ваша задача — восстановить прямоугольник, если это возможно. Заданные прямоугольники разрешается вращать по часовой стрелке на 90 градусов от 0 до 3 раз. Два маленьких прямоугольника в собранном прямоугольнике могут иметь общую сторону только в том случае, если эта сторона бесцветна. В собранном прямоугольнике должны быть использованы все заданные маленькие прямоугольники, ничего не должно остаться. Стороны собранного прямоугольника должны быть раскрашены в различные цвета, каждая сторона в свой цвет.

### Входные данные

Первая строка входного файла содержит одно целое число  $N$  — количество имеющихся маленьких прямоугольников ( $1 \leq N \leq 10^3$ ).

Следующие  $N$  строк задают маленькие прямоугольники. Каждый прямоугольник описывается на отдельной строке. Его описание состоит из последовательности, содержащей два целых числа  $w$  и  $h$ , за которыми идут четыре буквы  $c_0, c_1, c_2$  и  $c_3$ , записанные через пробел.  $w$  и  $h$  задают ширину и высоту прямоугольника ( $1 \leq w, h \leq 10^5$ ), а буквы — цвета сторон его в порядке обхода по часовой стрелке, начиная с верхней стороны прямоугольника.  $c_j \in \{\mathbf{R}, \mathbf{G}, \mathbf{B}, \mathbf{Y}, \mathbf{N}\}$  ( $0 \leq j \leq 3$ ), где  $\mathbf{R}, \mathbf{G}, \mathbf{B}, \mathbf{Y}$  — цвета сторон исходного прямоугольника, а  $\mathbf{N}$  обозначает отсутствие цвета.

### Выходные данные

Если восстановить прямоугольник, используя все заданные прямоугольники, невозможно, то в выходной файл нужно вывести единственную строку **IMPOSSIBLE**.

В противном случае необходимо вывести  $N$  строк, которые описывают положение заданных маленьких прямоугольников в собранном прямоугольнике в порядке их перечисления во входном файле. Совместим начало координат с левым нижним углом собранного прямоугольника, ось  $Ox$  пойдет по нижней его стороне, а ось  $Oy$  — по левой вертикальной стороне. Тогда каждая строка будет содержать описание положения одного маленького прямоугольника в результате. Описание должно состоять из трех целых чисел, записанных через пробел. Первые два числа  $x$  и  $y$  задают координаты левого нижнего угла соответствующего прямоугольника в собранном прямоугольнике, а третье число  $r$  — количество его вращений по часовой стрелке ( $0 \leq x, y \leq 10^5, 0 \leq r \leq 3$ ).

Если существует несколько способов собрать прямоугольник, то выведите любой.

### Примеры

<i>input.txt</i>	<i>output.txt</i>
2 3 4 R N B Y 4 4 R G B N	0 0 0 3 0 0
2 3 4 R B B Y 4 4 R G B B	<b>IMPOSSIBLE</b>
1 5 5 R G B N	<b>IMPOSSIBLE</b>
4 2 3 N N Y R 3 3 B Y N N 4 3 N G B N 4 2 R G N N	0 3 1 0 0 2 3 0 0 3 3 0



## II (муниципальный) этап Всероссийской олимпиады школьников 2011/2012 уч. год. Новосибирская область

Предмет	Класс	Дата	Время начала	Время окончания
<b>Информатика</b>	<b>7-11</b>	<b>24.11.2011</b>	<b>10:00</b>	<b>15:00</b>

### Задача 5. Волшебные комнаты

Вы попали в странный лабиринт, состоящий из комнат. Между любыми двумя комнатами изначально может быть коридор, но его может и не быть. Этот лабиринт не простой. В нём так же существуют «волшебные» комнаты. Попадая в такую комнату, человек может применить, если захочет, немного магии: все коридоры, которые существовали до этого, исчезнут, однако если между двумя комнатами раньше коридора не существовало, то он появится.

Вам необходимо по коридорам как можно быстрее добраться из некоторой начальной комнаты в заданную конечную комнату.

#### Входные данные

Первая строка входного файла содержит два целых числа  $N$  и  $M$  — количество комнат и количество коридоров соответственно ( $1 \leq N, M \leq 10^5$ ). Все комнаты в лабиринте пронумерованы числами от 1 до  $N$ .

Во второй строке записаны через пробел еще два целых числа  $S$  и  $F$  — номера начальной и конечной комнат ( $1 \leq S, F \leq N$ ).

Третья строка содержит число  $W$  — количество «волшебных» комнат ( $0 \leq W \leq N$ ).

В четвёртой строке записаны через пробел  $W$  различных целых чисел — номера волшебных комнат.

В следующих  $M$  строках приведено описание коридоров. Каждый коридор описывается на отдельной строке. Он задается двумя целыми числами, записанными через пробел, номерами комнат, которые этот коридор соединяет. Между любыми двумя комнатами существует не более одного коридора.

#### Выходные данные

В единственной строке выходного файла должно содержаться одно целое число — минимальное количество коридоров, которые придётся преодолеть, чтобы добраться из комнаты номер  $S$  в комнату с номером  $F$ . Если невозможно по заданному лабиринту пройти из комнаты  $S$  в комнату  $F$ , то в выходной файл следует вывести слово **IMPOSSIBLE**.

#### Пример

<i>input.txt</i>	<i>output.txt</i>
5 5 1 5 1 2 1 2 1 3 1 4 3 5 4 5	2