

## Общее описание задач

<b>№</b>	<b>Название задачи</b>	<b>Ограничение на объем используемой памяти</b>	<b>Ограничение на время выполнения программы, секунды</b>	<b>Количество баллов, присвоенных задаче</b>
1.	MoldSEF	$\leq 32 \text{ Mb}$	$\leq 1,0$	100
2.	Aria	$\leq 256 \text{ Mb}$	$\leq 1,0$	100
3.	Or	$\leq 256 \text{ Mb}$	$\leq 1,0$	100

*Примечание.* При равенстве общего количества очков, приоритет будет отдан участнику, набравшему первым соответствующее количество очков.

### MoldSEF

Известно, что каждый год Национальное Агентство по Учебным Программам и Оценке организует конкурс MoldSEF. Конкурс проводится по трем тематическим категориям: Науки, Прикладные науки и Информационные технологии. К большому удивлению организаторов, в этом году зарегистрировалось очень много участников, так что Агентству пришлось организовать предварительный отбор участников. Для упрощения учащиеся объединены в команды по двое. Очевидно, что есть ученики, которые хотят участвовать в нескольких категориях. Чтобы выбрать лучшие команды, Агентство организовало предварительный отбор следующим образом:

1. Каждый ученик проходит тест, который содержит вопросы по каждой из трех категорий.
2. Жюри оценивает каждого ученика и присуждает баллы (положительные целые числа) за каждую категорию. Ученик  $i$  будет иметь оценки  $A_i, B_i, C_i$ .
3. Считается, что два ученика  $i$  и  $j$  совместимы и могут сформировать команду в категории, если и только если они имеют совпадение в оценках только по одной категории, то есть из условий  $A_i = A_j, B_i = B_j, C_i = C_j, i < j$ , только одно является верным.
4. На финальный конкурс допускаются только команды, составленные из совместимых учеников. Один ученик может быть совместим с несколькими учениками.

**Задание.** Напишите программу, которая из  $N$  учеников вычисляет количество потенциальных совместимых команд (пар совместимых учеников) на основе результатов оценок.

**Входные данные.** Первая строка стандартного ввода содержит целое число  $N$  – количество учеников. Каждая из следующих  $N$  строк содержит три разделенных пробелом целых числа. Стока  $i$  содержит значения  $A_i, B_i, C_i$  – оценки ученика  $i$  по трем категориям.

**Выходные данные.** Первая и единственная строка стандартного вывода будет содержать количество потенциальных совместимых команд.

**Ограничения.**  $1 \leq N \leq 100\,000, 1 \leq A_i, B_i, C_i \leq 100, 1 \leq i \leq N$ . Ограничения, касающиеся времени выполнения и объема используемой памяти, приведены в общем описании задач, предложенных для решения. Исходный файл будет иметь название `moldsef.pas`, `moldsef.c` или `moldsef.cpp`.

#### Пример 1.

*Вход*

3
1 2 3
1 4 5
1 2 4

*Выход*

2
---

**Пояснения:** Две потенциальные команды могут быть сформированы из первого и второго ученика (совпадение оценок по первой категории) или из второго и третьего ученика (также совпадение оценок по первой категории).

**Пример 2.***Вход*

```
4
100 100 100
100 100 100
100 99 99
99 99 100
```

*Выход*

```
5
```

**Пояснения:** Пять потенциальных команд могут быть сформированы из:

Echipa 1 – elevii 1, 3 (coincidentă pe categoria 1)

Echipa 2 – elevii 1, 4 (coincidentă pe categoria 3)

Echipa 3 – elevii 2, 3 (coincidentă pe categoria 1)

Echipa 4 – elevii 2, 4 (coincidentă pe categoria 3)

Echipa 5 – elevii 3, 4 (coincidentă pe categoria 2)

**Оценивание.**

1. Для 45% тестов  $1 \leq N \leq 100$ ,  $1 \leq Ai, Bi, Ci \leq 50$
2. Остальные 55% без дополнительных ограничений ( $1 \leq N \leq 100\ 000$ ,  $1 \leq Ai, Bi, Ci \leq 100$ )

## Aria

Виктор устроился аниматором в летний лагерь для детей. Однажды он получил задание организовать мероприятие по рисованию. Для этого мероприятия у Виктора есть большой лист бумаги в клетку и большой набор фломастеров разных цветов. Чтобы вовлечь в активность всех  $N$  детей, Виктор рисует координатную ось, центрированную посередине листа, и распределяет каждому ребенку по фломастеру и размеры прямоугольника, указывая его ширину и высоту. Задача каждого ребенка - нарисовать и раскрасить прямоугольник указанных размеров, центрированный в центре координат, с шириной параллельной оси  $x$  и высотой параллельной оси  $y$ .

После завершения мероприятия Виктор задается вопросом, какова будет площадь закрашенной части листа бумаги, то есть количество единичных квадратов, которые принадлежат хотя бы одному прямоугольнику.

**Задание.** Напишите программу, которая определяет площадь, покрытую  $N$  прямоугольниками.

**Входные данные.** Первая строка стандартного ввода содержит целое число  $N$  – количество прямоугольников. Каждая из следующих  $N$  строк содержит два четных целых числа  $X$  и  $Y$ , разделенных пробелом, размеры (ширину и соответственно высоту) каждого прямоугольника.

**Выходные данные.** Первая и единственная строка стандартного вывода будет содержать вычисленную площадь.

Ограничения.  $1 \leq N \leq 1\,000\,000$ ,  $2 \leq X, Y \leq 10^7$ . Ограничения, касающиеся времени выполнения и объема используемой памяти, приведены в общем описании задач, предложенных для решения. Исходный файл будет иметь название `aria.pas`, `aria.c` или `aria.cpp`.

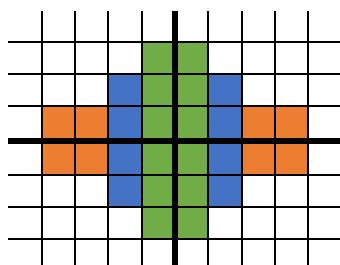
### Пример 1.

*Вход*

3
8 2
4 4
2 6

*Выход*

28
----



**Объяснение:** Три прямоугольника, раскрашенные в разные цвета, покрывают 28 единичных квадратов, то есть площадь составляет 28.

**Пример 2.**

*Вход*

```
5
2 10
4 4
2 2
8 8
6 6
```

*Выход*

```
68
```

**Оценка.** Тесты будут сгруппированы в следующие категории:

- A. Для 10% тестов: все прямоугольники находятся друг внутри друга.
- B. Для 35% тестов:  $X, Y < 3333$ .
- C. Для 45% тестов: ни один прямоугольник не будет расположен строго внутри другого прямоугольника.
- D. Для остальных 10% тестов: без дополнительных ограничений.

## Or

Виктор строит простую логическую схему в своей мастерской. Схема состоит из  $n$  стартовых проводов, обозначенных как  $x_1, x_2, \dots, x_n$  и  $m$  логических элементов  $OR$ , обозначенных как  $c_1, c_2, \dots, c_m$ . Каждый элемент имеет ровно два входа и один выход. Каждый из входов подключен либо к стартовому проводу  $x_j$ , либо к выходу другого элемента  $c_j$ . Конечно, в логической схеме нет циклов, и более того, соблюдается условие, что вход  $c_j$  может быть подключен к выходу  $c_i$  только если  $i < j$ .

Каждый стартовый провод в схеме может быть установлен в значение  $0$  или  $1$ , а значение выхода каждого элемента является логической операцией  $OR$  его входов - значение равно  $0$ , если значения обоих входов равны  $0$ , в противном случае  $-1$ .

Виктор не знает начальные значения стартовых проводов, но с помощью тщательных измерений он определил значения выходов некоторых элементов.

**Задача.** Напишите программу, которая определяет оставшиеся значения выходов, которые можно однозначно определить на основе измерений.

**Входные данные.** Первая строка стандартного ввода содержит положительные целые числа  $n$  и  $m$ , разделенные пробелом - количество стартовых проводов и количество элементов в схеме. Следующая строка содержит строку из  $m$  символов, описывающих измеренное значение выхода элемента  $c_j$ , или символ ‘?’ , если Виктор не проводил это измерение. Следующие  $m$  строк содержат метки входов. Стока  $j$ ,  $j \leq m$  содержит метки двух входов схемы  $c_j$ , каждая метка является либо проводом старта в форме " $x_i$ ", где соблюдается  $1 \leq i \leq n$ , либо элементом " $c_i$ ", где соблюдается  $1 \leq i < j$ . Два входа элемента  $c_j$  могут быть одинаковыми. Вы можете предположить, что измеренные значения не противоречивы.

**Выходные данные.** Первая и единственная строка стандартного вывода будет содержать строку из  $m$  символов - символ  $j$  в строке должен соответствовать значению выхода  $c_j$  или быть ‘?’ , если это значение не может быть определено однозначно.

**Ограничения.**  $2 \leq n \leq 10\ 000$ ;  $1 \leq m \leq 10\ 000$ . Ограничения по времени выполнения и объему используемой памяти даны в общем описании задач, предложенных для решения. Исходный файл будет иметь имя `or.pas`, `or.c` или `or.cpp`.

### Пример 1.

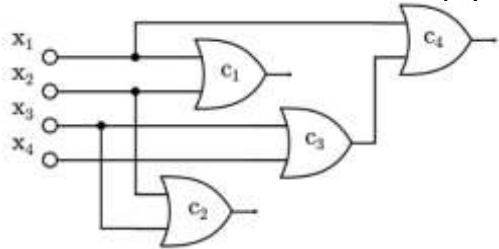
*Вход*

4 4
10??
x1 x2
x2 x3
x3 x4
x1 c3

*Выход*

10?1

**Объяснение:** Входные данные формируют следующую схему:



Мы видим, что  $c_1 = 1$ ,  $c_2 = 0$ , а значения  $c_3$  и  $c_4$  неизвестны.

Из  $c_1 = 1$  мы можем сделать вывод, что либо  $x_1$ , либо  $x_2$ , либо оба имеют значение 1. Также из  $c_2 = 0$  можно сделать вывод, что  $x_2$  и  $x_3$  имеют значение 0. Следовательно,  $x_1$  имеет значение 1. Значение  $c_3$  не может быть определено, поскольку значение  $x_3$  равно 0, следовательно,  $c_3$  может быть 0 или 1, поскольку  $x_4$  может быть 0 или 1. Поскольку значение  $x_1$  равно 1, мы делаем вывод, что  $c_4$  равно 1. Получаем ответ 10?1.

### Пример 1.

*Вход*

```

4 5
11???
x1 x2
x3 x4
x1 x3
x2 x4
c3 c4
  
```

*Выход*

```

11??1
  
```

**Оценка.** Тесты будут сгруппированы в следующие категории, как указано ниже:

- A. Для 7% тестов,  $n \leq 15$ ,  $m \leq 20$ .
- B. Для других 42% тестов,  $n \leq 500$ ,  $m \leq 500$ .
- C. Для оставшихся 51% тестов нет дополнительных ограничений.