

## Содержание

Задача А. Соединение и разъединение [3 секунды, 512 мегабайт]	2
Задача В. Сокровища [2 секунды, 256 мегабайт]	3
Задача С. Мэйвис и Дракула [2 секунды, 256 мегабайт]	4
Задача D. Конфеты и Камни [15 секунд, 50 мегабайт]	5
Задача Е. Work for Robots [2 секунды, 256 мегабайт]	6
Задача F. Классные парты [2 секунды, 512 мегабайт]	7
Задача G. Мэйвис в школе [3 секунды, 256 мегабайт]	9

---

### Задача А. Соединение и разъединение [3 секунды, 512 мегабайт]

Вы когда-нибудь слышали про обход в глубину? Например, используя этот алгоритм, вы можете проверить является ли граф связным за время  $O(E)$ . Вы можете даже посчитать количество компонент связности за то же время.

А вы когда-нибудь слышали про систему непересекающихся множеств? Используя эту структуру, вы можете быстро обрабатывать запросы “Добавить ребро в граф” и “Посчитать количество компонент связности в графе”.

А вы когда-нибудь слышали о *динамической* задаче связности? В этой задаче вам необходимо обрабатывать три типа запросов:

1. Добавить ребро в граф.
2. Удалить ребро из графа.
3. Посчитать количество компонент связности в графе.

Можно считать, что граф является неориентированным. Изначально граф является пустым.

#### Формат входных данных

В первой строке находятся два целых числа  $N$  и  $K$  — количество вершин и количество запросов, соответственно ( $1 \leq N \leq 300\,000$ ,  $0 \leq K \leq 300\,000$ ). Следующие  $K$  строк содержат запросы, по одному в строке. Каждый запрос имеет один из трех типов:

1.  $+ \ u \ v$ : Добавить ребро между вершинами  $u$  и  $v$ . Гарантируется, что такого ребра нет.
2.  $- \ u \ v$ : Удалить ребро между  $u$  и  $v$ . Гарантируется, что такое ребро есть.
3.  $?$ : Посчитать количество компонент связности в графе.

Вершины пронумерованы целыми числами от 1 до  $N$ . Во всех запросах  $u \neq v$ .

#### Формат выходных данных

Для каждого запроса типа ‘?’, Выведите количество компонент связности в момент запроса.

#### Примеры

стандартный ввод	стандартный вывод
5 11	5
?	1
+ 1 2	1
+ 2 3	2
+ 3 4	
+ 4 5	
+ 5 1	
?	
- 2 3	
?	
- 4 5	
?	

### Задача В. Сокровища [2 секунды, 256 мегабайт]

Дочь короля Флатландии собирается выйти за прекрасного принца. Принц хочет подарить принцессе сокровища, но он не уверен какие именно бриллианты из своей коллекции выбрать.

В коллекции принца  $n$  бриллиантов, каждый характеризуется весом  $w_i$  и стоимостью  $v_i$ . Принц хочет подарить наиболее дорогие бриллианты, однако король умен и не примет бриллиантов суммарного веса больше  $R$ . С другой стороны, принц будет считать себя жадным всю оставшуюся жизнь, если подарит бриллиантов суммарным весом меньше  $L$ .

Помогите принцу выбрать набор бриллиантов наибольшей суммарной стоимости, чтобы суммарный вес был в отрезке  $[L, R]$ .

#### Формат входных данных

Первая строка содержит число  $n$  ( $1 \leq n \leq 32$ ),  $L$  и  $R$  ( $0 \leq L \leq R \leq 10^{18}$ ). Следующие  $n$  строк описывают бриллианты и содержит по два числа — вес и стоимость соответствующего бриллианта ( $1 \leq w_i, v_i \leq 10^{15}$ ).

#### Формат выходных данных

Первая строка вывода должна содержать  $k$  — количество бриллиантов, которые нужно подарить принцессе. Вторая строка должна содержать номера даримых бриллиантов.

Бриллианты нумеруются от 1 до  $n$  в порядке появления во входных данных.

Если составить подарок принцессе невозможно, то выведите 0 в первой строке вывода.

#### Примеры

стандартный ввод	стандартный вывод
3 6 8	1
3 10	2
7 3	
8 2	

### Задача С. Мэйвис и Дракула [2 секунды, 256 мегабайт]

Сегодня один из тех вечеров, когда Мэйвис и Дракула проводят время вместе. Дракула решил поиграть с дочкой в игру. Она, конечно же, как вы уже наверное догадались, имеет математическую основу.

Состоит эта игра в следующем: Дракула говорит Мэйвис натуральные числа  $A$  и  $B$ , после чего Мэйвис должна найти такое натуральное число  $x$ , не превосходящее  $B$ , что  $\text{lcm}(A, x)$  максимально среди всех натуральных чисел, не превосходящих  $B$ . Дракула хочет сыграть с дочкой  $T$  раундов.

Мэйвис достаточно быстро поднадоела эта математическая игра, и она решила попросить вас написать программу, которая по заданным  $A$  и  $B$  будет находить число наибольшее значение  $\text{lcm}$ , которого можно достигнуть.

#### Формат входных данных

В первой строке содержится одно натуральное число  $T$  ( $1 \leq T \leq 10^5$ ) — количество раундов в игре. В следующих  $T$  строках содержатся числа  $A$  и  $B$  ( $1 \leq A, B \leq 3 \times 10^9$ ) для каждого раунда.

#### Формат выходных данных

В  $T$  строках должны содержаться ответы для раундов — наибольшее значение  $\text{lcm}$ , которого можно достигнуть в этом раунде.

#### Примеры

стандартный ввод	стандартный вывод
6	190
10 20	260
20 13	24
8 4	561
17 34	44
4 11	342
18 20	

### Задача D. Конфеты и Камни [15 секунд, 50 мегабайт]

Мальчик Геральд и его тренер Миша играют в интересную игру. В начале игры имеются куча из  $n$  конфет и куча из  $m$  камней. Геральд и Миша ходят по очереди, первым ходит Миша. Миша на своем ходу проверяет, сколько на данный момент Геральд съел конфет и камней. Пусть Геральд съел  $a$  конфет и  $b$  камней. Тогда Миша начисляет Геральду  $f(a, b)$  призовых очков. Геральд же на своем ходу съедает либо одну конфету из кучи с конфетами, либо один камень из кучи с камнями. Когда Миша обнаруживает, что Геральд съел все, кроме одной конфеты и одного камня, он последний раз начисляет очки и игра заканчивается. Опустошать ни ту, ни другую кучку Геральд не имеет права. Расскажите Геральду, как ему играть, чтобы получить наибольшее количество очков: требуется найти один из возможных оптимальных вариантов игры для Геральда.

#### Формат входных данных

В первой строке содержатся три целых числа  $n, m, p$  ( $1 \leq n, m \leq 20000$ ,  $1 \leq p \leq 10^9$ ). Во второй строке находятся  $n$  целых чисел  $x_0, x_1, \dots, x_{n-1}$  ( $0 \leq x_i \leq 20000$ ). В третьей строке находятся  $m$  целых чисел  $y_0, y_1, \dots, y_{m-1}$  ( $0 \leq y_i \leq 20000$ ). Величина  $f(a, b)$  вычисляется, как остаток от деления суммы  $x_a + y_b$  на число  $p$ .

#### Формат выходных данных

В первой строке выведите единственное число: максимальное количество призовых очков, которое может заработать Геральд. Во второй строке выведите строку из  $n + m - 2$  символов, каждый из которых — это «C» или «S»,  $i$ -ый символ должен быть «C», если на своем  $i$ -ом ходу Геральд должен съесть конфету, и «S», если камень.

#### Примеры

стандартный ввод	стандартный вывод
2 2 10 0 0 0 1	2 SC
3 3 10 0 2 0 0 0 2	10 CSSC
3 3 2 0 1 1 1 1 0	4 SCSC

#### Замечание

В первом тесте, если на первом ходу Геральд съест камень, то после него он получит одно очко, а если конфету — то ноль. Перед первым своим ходом Геральд получит в любом случае 0 очков, а после второго — в любом случае 1. Таким образом, максимальное количество очков, которое может получить Геральд равно 2, и для этого надо съесть сначала камень, потом конфету.

### Задача Е. Work for Robots [2 секунды, 256 мегабайт]

На планете PTZZZ живёт и работает  $n$  роботов. С незапамятных времён некоторые роботы дружат между собой, а некоторые — нет. Ровно один раз в день некоторые роботы выходят на работу, а все остальные идут в парк развлечений и отдыхают. При этом на работу должен выйти хотя бы один робот. Определением того, кто будет работать, а кто отдыхать, занимается робот-директор. Работа настолько важна для роботов, что первый день, когда робот-директор принял решение, был назван Первым днём Мира. Если окажется так, что в некоторый день на работу вышли в точности те же роботы, что и в какой-нибудь из предыдущих дней, то робот-директор заржавеет от горя. Кроме того, закон не позволяет роботу-директору отправлять на работу группу роботов, в которой какие-нибудь два робота не будут дружить между собой. Робот-директор не хочет ржаветь от горя как можно дольше, поэтому он старается каждый раз отправлять на работу другую группу роботов. Однако, как легко понять, различных групп роботов всё же конечное число, поэтому роботу-директору в конце концов придётся заржаветь. От Вас требуется найти лишь номер дня, в который это произойдёт.

#### Формат входных данных

В первой строке записано целое число  $n$ , количество роботов на PTZZZ ( $1 \leq n \leq 50$ ). В каждой из следующих  $n$  строк записано по  $n$  цифр 0 или 1.  $j$ -я цифра в  $i$ -й строке равна единице, если  $i$ -й и  $j$ -й роботы дружат между собой, и нулю в противном случае. Гарантируется, что  $i$ -я цифра в  $i$ -й строке равна нулю, а  $j$ -я цифра в  $i$ -й строке совпадает с  $i$ -й цифрой в  $j$ -й строке.

#### Формат выходных данных

Выведите номер дня, в который робот-директор вынужден будет заржаветь от горя.

#### Пример

стандартный ввод	стандартный вывод
6 011100 101100 110100 111000 000001 000010	19

## Задача F. Классные парты [2 секунды, 512 мегабайт]

Для нового кабинета школы Иннополиса требуется купить  $n$  двухместных парт.

Парты бывают  $k$  типов, которые задают их размер. Парты типа  $i$  подходят школьникам, рост которых находится в диапазоне от  $L_i$  до  $R_i$  включительно. Остальным школьникам сидеть за такой партой неудобно, при этом *величиной неудобства* школьника, если он сидит за такой партой, будем называть модуль разности его роста и ближайшей границы диапазона этой парты. Если парты школьнику подходит, то для него величина неудобства равна нулю.

Например, если  $L_i = 100$  и  $R_i = 120$ , то неудобство для школьника с ростом 80 равно 20, для школьника с ростом 130 равно 10, а для школьника с ростом 105 равно 0.

В кабинете по очереди будут заниматься  $m$  групп школьников, каждая из которых состоит из  $2n$  человек. Известен рост каждого школьника в каждой из групп. Закупленные парты будут расставлены в классе, и в каждой группе за каждой партой будут сидеть ровно два школьника. Необходимо купить  $n$  парт и рассадить за ними школьников каждой группы таким образом, чтобы суммарное неудобство для всех школьников, занимающихся в этом кабинете, было минимальным.

Требуется написать программу, которая по информации о каждом из  $k$  типов парт и известным значениям роста каждого школьника в каждой группе определяет, какого минимального суммарного значения неудобства школьников можно достичь, купив парты и рассадив за них школьников в каждой группе оптимальным образом.

### Формат входных данных

В первой строке входных данных находятся три целых числа  $m$ ,  $n$  и  $k$  ( $1 \leq m, n \leq 200\,000$ ;  $1 \leq m \cdot n \leq 200\,000$ ;  $2 \leq k \leq 200\,000$ ) — количество групп школьников, которые будут заниматься в кабинете, количество парт, которые необходимо купить, и количество типов парт соответственно.

В каждой из следующих  $k$  строк находятся по два целых числа  $L_i$  и  $R_i$  ( $1 \leq L_i \leq R_i \leq 10^9$ ), характеризующие диапазон роста школьников, для которых подходят парты типа  $i$ .

В каждой из следующих  $m$  строк находится описание группы. Каждое описание состоит из  $2n$  целых чисел  $h_1, h_2, \dots, h_{2n}$ , задающих значение роста каждого из  $2n$  школьников группы ( $1 \leq h_i \leq 10^9$ ).

### Формат выходных данных

В единственной строке выходных данных выведите  $P$  — минимальную величину суммарного неудобства, которую можно достичь при оптимальной покупке парт.

### Примеры

стандартный ввод	стандартный вывод
1 2 2 5 25 50 90 60 5 10 40	10
2 3 3 200 400 300 500 100 600 300 330 440 40 30 300 150 250 350 450 550 300	130
1 3 4 10 100 200 200 10 100 300 1000 5 10 20 15 200 90	105

### Пояснение к примеру

В первом примере есть только одна группа школьников, занимающаяся в классе. Следует купить по одной парте каждого вида и посадить школьников с ростами 5 и 10 за парту первого типа, а школьников с ростами 40 и 60 за парту второго типа. В таком случае неудобно сидеть будет только школьнику с ростом 40 и соответствующая величина неудобства будет равна 10.



### Задача G. Мэйвис в школе [3 секунды, 256 мегабайт]

Когда-то давным давно Мэйвис училась в вампирской школе. На уроках информатики Мэйвис решала задачки на различные темы. Одна из ее любимых тем — это массивы.

И вот наступил важный день: в вампирской школе контрольная по этой теме. На контрольной ей досталась следующая задача: дан массив  $a_i$ , нужно найти в нем отрезок  $[l..r]$  с максимальной величиной  $\max(l..r) \times \text{xor}(l..r)$ .

$\max(l..r)$  — это максимальное число на отрезке  $[l..r]$ , а  $\text{xor}(l..r)$  — это  $\oplus$ -сумма чисел на этом отрезке.

Помогите ей решить эту задачу, ведь она очень не хочет огорчать Дракулу.

#### Формат входных данных

В первой строке находятся натуральное число  $n$  ( $1 \leq n \leq 10^5$ ), в следующей строке находятся  $n$  чисел  $a_i$  ( $1 \leq a_i \leq 10^6$ ) — массив, который есть у Мэйвис.

#### Формат выходных данных

В первой строке выведите целое число — наибольшее произведение  $\max(l..r) \times \text{xor}(l..r)$ , которое можно найти в массиве.

#### Пример

стандартный ввод	стандартный вывод
4 1 2 10 7	150
6 2 3 5 3 6 8	112

#### Замечание

В первом примере ответом является отрезок  $[2..4]$ .

Во втором примере ответом является отрезок  $[5..6]$ .