

## Задача А. Кубики

Имя входного файла: `cubes.in`  
Имя выходного файла: `cubes.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Родители подарили Сене набор детских кубиков. Поскольку Сеня скоро пойдет в школу, они купили ему кубики с буквами. На каждой из шести граней каждого кубика написана буква.

Теперь Сеня хочет похвастаться перед старшей сестрой, что научился читать. Для этого он хочет сложить из кубиков ее имя. Но это оказалось довольно сложно сделать — ведь разные буквы могут находиться на одном и том же кубике и тогда Сеня не сможет использовать обе буквы в слове. Правда одна и та же буква может встречаться на разных кубиках. Помогите Сене!

Дан набор кубиков и имя сестры. Выясните, можно ли выложить ее имя с помощью этих кубиков и если да, то в каком порядке следует выложить кубики.

### Формат входных данных

На первой строке входного файла находится число  $n$  ( $1 \leq n \leq 100$ ) — количество кубиков в наборе у Сени. На второй строке записано имя Сениной сестры — слово, состоящие только из больших латинских букв, не длиннее 100 символов. Следующие  $n$  строк содержат по 6 букв (только большие латинские буквы), которые написаны на соответствующем кубике.

### Формат выходных данных

На первой строке выходного файла выведите «YES», если выложить имя Сениной сестры данными кубиками можно, «NO» в противном случае.

Если ответ «YES», на второй строке выведите  $m$  различных чисел из диапазона от 1 до  $n$ , где  $m$  — количество букв в имени Сениной сестры,  $i$ -е число должно быть номером кубика, который следует положить на  $i$ -е место при составлении имени Сениной сестры. Кубики нумеруются с 1, в том порядке, в котором они заданы во входном файле. Если решений несколько, выведите любое. Разделяйте числа пробелами.

### Примеры

<code>cubes.in</code>	<code>cubes.out</code>
4 ANN ANNNNN BCDEFG HIJKLM NOPQRS	NO
5 HELEN ABCDEF GHIJKL MNOPQL STUVWN EIUOZK	YES 2 5 3 1 4

## Задача В. Эйлеров путь

Имя входного файла: **euler.in**  
Имя выходного файла: **euler.out**  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дан неориентированный связный граф, не более трех вершин имеет нечетную степень. Требуется определить, существует ли в нем путь, проходящий по всем ребрам.

Если такой путь существует, необходимо его вывести.

### Формат входных данных

Первая строка входного файла содержит натуральное число  $n$  — количество вершин графа ( $1 \leq n \leq 100\,000$ ). Далее следуют  $n$  строк, задающих ребра. В  $i$ -й из этих строк находится число  $m_i$  — количество ребер, инцидентных вершине  $i$ . Далее следуют  $m_i$  натуральных чисел — номера вершин, в которые ведут ребро из  $i$ -й вершины.

Граф может содержать кратные ребра, но не содержит петель.

Граф содержит не более 300 000 ребер.

### Формат выходных данных

Если решение существует, то в первую строку выходного файла выведите одно число  $k$  — количество ребер в искомом маршруте, а во вторую  $k + 1$  число — номера вершин в порядке их посещения.

Если решений нет, выведите в выходной файл одно число **-1**.

Если решений несколько, выведите любое.

### Примеры

euler.in	euler.out
4 2 2 2 4 1 4 3 1 2 2 4 2 3 2	5 1 2 3 4 2 1

## Задача С. Замощение доминошками

Имя входного файла: dominoes.in  
Имя выходного файла: dominoes.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Дано игровое поле размера  $n \times m$ , некоторые клетки которого уже замощены. Замостить свободные соседние клетки поля доминошкой размера  $1 \times 2$  стоит  $a$  условных единиц, а замостить свободную клетку поля квадратиком размера  $1 \times 1$  —  $b$  условных единиц.

Определите, какая минимальная сумма денег нужна, чтобы замостить всё поле.

### Формат входных данных

Первая строка входного файла содержит 4 целых числа  $n, m, a, b$  ( $1 \leq n, m \leq 100, |a| \leq 1000, |b| \leq 1000$ ). Каждая из последующих  $n$  строк содержит по  $m$  символов: символ ‘‘.’’ (точка) обозначает занятую клетку поля, а символ ‘‘\*’’ (звёздочка) — свободную.

### Формат выходных данных

В выходной файл выведите одно число — минимальную сумму денег, имея которую можно замостить свободные клетки поля (и только их).

### Пример

dominoes.in	dominoes.out
2 3 3 2 .** . *	5

## Задача D. День рождения

Имя входного файла: **birthday.in**  
Имя выходного файла: **birthday.out**  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Митя знаком с  $m$  юношами и  $n$  девушками и хочет пригласить часть из них на свой день рождения. Ему известно, с какими девушками знаком каждый юноша, и с какими юношами знакома каждая девушка. Он хочет добиться того, чтобы каждый приглашённый был знаком со всеми приглашёнными противоположного пола, пригласив при этом максимально возможное число своих знакомых. Помогите ему это сделать!

### Формат входных данных

Входной файл состоит из одного или нескольких наборов входных данных. В первой строке входного файла записано число наборов  $k$  ( $1 \leq k \leq 20$ ). В последующих строках записаны сами наборы входных данных.

В первой строке каждого набора задаются числа  $0 \leq m \leq 150$  и  $0 \leq n \leq 150$ . Далее следуют  $m$  строк, в каждой из которых записано одно или несколько чисел — номера девушек, с которыми знаком  $i$ -й юноша (каждый номер встречается не более одного раза). Стока завершается числом 0.

### Формат выходных данных

Для каждого набора выведите четыре строки. В первой из них выведите максимальное число знакомых, которых сможет пригласить Митя. В следующей строке выведите количество юношей и количество девушек в максимальном наборе знакомых. Следующие две строки должны содержать номера приглашённых юношей и приглашённых девушек соответственно. Если максимальных наборов несколько, то выведите любой из них.

### Примеры

birthday.in	birthday.out
2	4
2 2	2 2
1 2 0	1 2
1 2 0	1 2
3 2	4
1 2 0	2 2
2 0	1 3
1 2 0	1 2

## Задача Е. Студентам — бесплатно!

Имя входного файла: **students-free.in**  
Имя выходного файла: **students-free.out**  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 64 мегабайта

Зал Большого галактического театра состоит из  $S$  рядов, по  $S$  мест в каждом ряду. Продажа билетов на каждый спектакль происходит по следующему принципу: первые  $S^2 - N$  ценителей прекрасного приобретают билеты на любые места по их вкусу, а оставшиеся  $N$  кресел администрация бесплатно выделяет студентам, отдавая дань сложившимся традициям.

Во избежание обвинений в дискриминации по половому признаку, рассаживать студентов по этим  $N$  местам необходимо таким образом, чтобы:

- в каждом ряду количество девушек-студенток и количество юношей-студентов различалось бы не более чем на 1;
- на каждой "вертикали мест" (т. е. местах, имеющих один и тот же номер, но расположенных в разных рядах) количество девушек-студенток и количество юношей-студентов также различалось бы не более чем на 1.

Таким образом, после продажи билетов ценители прекрасного билетёры должны распределить оставшиеся  $N$  кресел на женские и мужские с соблюдением этих правил. Каждое место в зале определяется двумя числами от 1 до  $S$  — номером ряда и номером самого места в этом ряду. Студенческое кресло номер  $i$  расположено в  $a_i$ -м ряду и имеет в нём номер  $b_i$ . Поскольку ценители прекрасного могли занять совершенно любые места, числа  $a_i$  и  $b_i$  могут принимать любые значения от 1 до  $S$ . В частности, может оказаться так, что в каком-нибудь ряду не будет ни одного студенческого места.

Ради упрощения работы билетёров администрация обращается к вам с заданием написать программу, которая автоматизирует процесс распределения студенческих мест на мужские и женские.

### Формат входных данных

Сначала вводятся два целых числа  $S$  и  $N$  ( $1 \leq S \leq 100\,000, 1 \leq N \leq \min\{100\,000, S^2\}$ ). Далее расположены  $N$  пар натуральных чисел  $(a_i, b_i)$ , не превосходящих  $S$ . Гарантируется, что все места различные.

### Формат выходных данных

Если искомого способа не существует, выведите **Impossible**. Иначе выведите единственную строку из  $N$  символов **M** (мужское) и **W** (женское). Символ на  $i$ -й позиции соответствует статусу  $i$ -го места в той же нумерации, в которой они были перечислены во входных данных.

### Примеры

students-free.in	students-free.out
2 2	WM
2 1	
1 2	
3 5	MMWWM
1 2	
2 3	
1 3	
2 1	
1 1	

## Задача F. Женитьба

Имя входного файла: **marriage.in**  
Имя выходного файла: **marriage.out**  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Давным давно в одной далёкой стране правил мудрый царь. И было у него ни много, ни мало  $M$  дочерей. Вот настало время выдавать дочерей замуж, и послал царь гонцов в  $N$  соседних государств. На эту весть съехалось по одному принцу от каждого государства. Так как царь был любящим отцом, учитывающим мнение своих дочерей, первым делом он потребовал принцев выстроиться в ряд, занумеровав юношей числами от 1 до  $N$ , и спросил у каждой дочери, с какими из стоящих молодых людей она согласна сыграть свадьбу. У царя этой страны было хорошее математическое образование, и ему не составило бы труда по этой информации проверить, можно ли назначить каждой дочери своего жениха из числа симпатичных ей молодых людей. Но пытливый ум правителя страны заинтересовал такой вопрос: сколько существует пар  $(L, R)$  ( $1 \leq L \leq R \leq N$ ), таких, что из юношей с номерами от  $L$  до  $R$  включительно можно найти по жениху для каждой из дочерей? Помогите царю найти ответ на его вопрос!

### Формат входных данных

В первой строке входного файла заданы три целых числа  $N$ ,  $M$  и  $K$  ( $1 \leq N \leq 30\,000$ ,  $1 \leq M \leq 2\,000$ ,  $1 \leq K \leq \min(N \cdot M, 100\,000)$  – соответственно количество юношей, количество девушек и количество строк, описывающих предпочтения девушек. В каждой из следующих  $K$  строк записаны два целых чисел  $A_i$ ,  $B_i$  ( $1 \leq A_i \leq N$ ,  $1 \leq B_i \leq M$ ), которые означают, что девушке  $B_i$  нравится юноша  $A_i$ . Все записи различны.

### Формат выходных данных

### Примеры

marriage.in	marriage.out
8 4 16 5 2 2 1 8 4 3 3 3 4 1 2 2 2 3 1 7 2 7 3 4 4 3 2 6 3 4 3 8 1 6 4	14