

## Лабораторная работа 2-4. Потoki и паросочетания

Statement is not available on English language

### А. Просто поток

ограничение по времени на тест: 5 секунд

ограничение по памяти на тест: 256 мегабайт

**ввод: стандартный ввод**

**вывод: стандартный вывод**

Дана система из узлов и труб, по которым может течь вода. Для каждой трубы известна наибольшая скорость, с которой вода может протекать через нее. Известно, что вода течет по трубам таким образом, что за единицу времени в каждый узел (за исключением двух — источника и стока) втекает ровно столько воды, сколько из него вытекает.

Ваша задача — найти наибольшее количество воды, которое за единицу времени может протекать между источником и стоком, а также скорость течения воды по каждой из труб.

Трубы являются двусторонними, то есть вода в них может течь в любом направлении. Между любой парой узлов может быть более одной трубы.

#### Входные данные

В первой строке записано натуральное число  $N$  — количество узлов в системе ( $2 \leq N \leq 100$ ). Известно, что источник имеет номер 1, а сток номер  $N$ . Во второй строке записано натуральное  $M$  ( $1 \leq M \leq 5000$ ) — количество труб в системе. Далее в  $M$  строках идет описание труб. Каждая труба задается тройкой целых чисел  $A_i, B_i, C_i$ , где  $A_i, B_i$  — номера узлов, которые соединяет данная труба ( $A_i \neq B_i$ ), а  $C_i$  ( $0 \leq C_i \leq 10^4$ ) — наибольшая допустимая скорость течения воды через данную трубу.

#### Выходные данные

В первой строке выведите наибольшее количество воды, которое протекает между источником и стоком за единицу времени. Далее выведите  $M$  строк, в каждой из которых выведите скорость течения воды по соответствующей трубе. Если направление не совпадает с порядком узлов, заданным во входных данных, то выводите скорость со знаком минус. Числа выводите с точностью  $10^{-3}$ .

### Примеры

входные данные	Copy
2 2 1 2 1 2 1 3	
выходные данные	Copy
4 1 -3	

Statement is not available on English language

## В. Разрез

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

**ввод: стандартный ввод**

**вывод: стандартный вывод**

Найдите минимальный разрез между вершинами 1 и  $n$  в заданном неориентированном графе.

### Входные данные

На первой строке входного файла содержится  $n$  ( $2 \leq n \leq 100$ ) — число вершин в графе и  $m$  ( $0 \leq m \leq 400$ ) — количество ребер. На следующих  $m$  строках входного файла содержится описание ребер. Ребро описывается номерами вершин, которые оно соединяет, и его пропускной способностью (положительное целое число, не превосходящее 10 000 000), при этом никакие две вершины не соединяются более чем одним ребром.

### Выходные данные

На первой строке выходного файла должны содержаться количество ребер в минимальном разрезе и их суммарная пропускная способность. На следующей строке выведите возрастающую последовательность номеров ребер (ребра нумеруются в том порядке, в каком они были заданы во входном файле).

### Примеры

<b>входные данные</b>	Copy
<pre>3 3 1 2 3 1 3 5 3 2 7</pre>	
<b>выходные данные</b>	Copy
<pre>2 8 1 2</pre>	

Statement is not available on English language

## С. Улиточки

ограничение по времени на тест: 2 секунды  
ограничение по памяти на тест: 256 мегабайт

**ввод: стандартный ввод**

**вывод: стандартный вывод**

Две улиточки Маша и Петя сейчас находятся в на лужайке с абрикосами и хотят добраться до своего домика. Лужайки пронумерованы числами от 1 до  $n$  и соединены дорожками (может быть несколько дорожек соединяющих две лужайки, могут быть дорожки, соединяющие лужайку с собой же). В виду соображений гигиены, если по дорожке проползла улиточка, то вторая по той же дорожке уже ползти не может. Помогите Пете и Маше добраться до домика.

### Входные данные

В первой строке файла записаны четыре целых числа —  $n$ ,  $m$ ,  $s$  и  $t$  (количество лужаек, количество дорог, номер лужайки с абрикосами и номер домика). В следующих  $m$  строках записаны пары чисел. Пара чисел  $(x, y)$  означает, что есть дорожка с лужайки  $x$  до лужайки  $y$  (из-за особенностей улиток и местности дорожки односторонние). Ограничения:  $2 \leq n \leq 10^5, 0 \leq m \leq 10^5, s \neq t$ .

## Выходные данные

Если существует решение, то выведите YES и на двух отдельных строчках сначала последовательность лужаек для Машеньки (дам нужно пропускать вперед), затем путь для Пети. Если решения не существует, выведите NO. Если решений несколько, выведите любое.

## Примеры

входные данные	Copy
3 3 1 3 1 2 1 3 2 3	
выходные данные	Copy
YES 1 3 1 2 3	

## Примечание

Дан оргграф, найти два непересекающихся по ребрам пути из  $s$  в  $t$ , вывести вершины найденных путей.

Statement is not available on English language

## D. Паросочетание

ограничение по времени на тест: 2 секунды  
ограничение по памяти на тест: 256 мегабайт

**ввод: стандартный ввод**

**вывод: стандартный вывод**

Двудольным графом называется неориентированный граф  $(V, E)$ ,  $E \subseteq V \times V$  такой, что его множество вершин  $V$  можно разбить на два множества  $A$  и  $B$ , для которых  $\forall (e_1, e_2) \in E, e_1 \in A, e_2 \in B$  и  $A \cup B = V, A \cap B = \emptyset$ .

Паросочетанием в двудольном графе называется любой набор его несмежных рёбер, то есть такой набор  $S \subseteq E$ , что для любых двух рёбер  $e_1 = (u_1, v_1), e_2 = (u_2, v_2)$  из  $S$   $u_1 \neq u_2$  и  $v_1 \neq v_2$ .

Ваша задача — найти максимальное паросочетание в двудольном графе, то есть паросочетание с максимально возможным числом рёбер.

### Входные данные

В первой строке записаны два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 250$ ), где  $n$  — число вершин в множестве  $A$ , а  $m$  — число вершин в  $B$ .

Далее следуют  $n$  строк с описаниями рёбер —  $i$ -я вершина из  $A$  описана в  $(i + 1)$ -й строке файла. Каждая из этих строк содержит номера вершин из  $B$ , соединённых с  $i$ -й вершиной  $A$ . Гарантируется, что в графе нет кратных ребер. Вершины в  $A$  и  $B$  нумеруются независимо (с единицы). Список завершается числом 0.

### Выходные данные

Первая строка выходного файла должна содержать одно целое число  $l$  — количество рёбер в максимальном паросочетании. Далее следуют  $l$  строк, в каждой из которых должны быть два целых числа  $u_j$  и  $v_j$  — концы рёбер паросочетания в  $A$  и  $B$  соответственно.

### Пример

#### входные данные

[Copy](#)

```
2 2
1 2 0
2 0
```

#### выходные данные

[Copy](#)

```
2
1 1
2 2
```

Statement is not available on English language

## Е. Шахматная доска

ограничение по времени на тест: 2 секунды

ограничение по памяти на тест: 256 мегабайт

**ввод: стандартный ввод**

**вывод: стандартный вывод**

Вася любит играть в необычные шахматы. Его братишка Коля был еще очень маленький. Как-то раз, когда Вася вернулся из школы, он увидел, что его любимую шахматную доску кто-то перекрасил. Вася не сильно разозлился на Колю, потому что очень любил своего брата. Так как у них дома были только черная и белая краски, каждая клетка доски была покрашена в один из этих двух цветов.

Вася решил исправить ошибку брата, он решил покрасить доску так, чтобы она снова стала шахматной. Но Вася почему-то подумал, что хочет красить только диагонали. Причем Вася решил не тратить много времени, поэтому его интересует способ покраски, который содержит наименьшее количество действий. За одно действие Вася может покрасить полностью какую-либо диагональ, в любой из двух цветов: черный или белый. Диагонали бывают двух типов, в зависимости от направления прямой, на которой лежит диагональ. Диагональ, которая лежит на прямой, направленной влево и вниз, является диагональю первого типа, а диагональ, которая лежит на прямой, направленной вправо и вниз, — второго.

Вам предстоит помочь Васе. Задано испорченное Колей шахматное поле. Вам необходимо определить, за какое минимальное количество действий Вася сможет перекрасить доску так, чтобы она стала шахматной.

### Входные данные

В первой строке входного файла записаны два целых числа:  $n$  и  $m$  ( $1 \leq n, m \leq 100$ ) — количество строчек и количество столбцов шахматного поля соответственно.

В следующих  $n$  строках записано поле, в каждой строке по  $m$  символов. Каждая строка входного файла описывает одну строку шахматного поля. W соответствует белой клетке, B — черной.

### Выходные данные

В выходной файл нужно вывести число  $p$ , количество действий, которое потребуется Васе, чтобы его доска снова стала шахматной. В следующих  $p$  строках описаны действия. Каждое действие описано тремя параметрами: тип диагонали, координаты клетки и цвет. Тип диагонали — это число 1 или 2. Координаты клетки — это два целых числа: строка и столбец одной из клеток, которую покрасили этим действием. Цвет — это символ W или B, белый и черный соответственно.

### Примеры

входные данные	Copy
3 3 WBB BWB BBW	
выходные данные	Copy
1 1 1 3 W	

<b>входные данные</b>	Copy
3 3 WBW WWB WWW	
<b>выходные данные</b>	Copy
1 2 2 1 B	

<b>входные данные</b>	Copy
1 3 WWW	
<b>выходные данные</b>	Copy
1 1 1 2 B	

Statement is not available on English language

## Г. Двигаем предметы

ограничение по времени на тест: 2 секунды  
ограничение по памяти на тест: 512 мегабайт

**ввод: стандартный ввод**

**вывод: стандартный вывод**

На плоскости расположены  $n$  предметов, их нужно переместить в заданные  $n$  позиций. При этом не важно, какой предмет какую из них займет. Для каждого предмета известна максимальная скорость, с которой его можно перемещать, при этом перемещать все предметы можно одновременно.

Найдите минимальное время, за которое можно переместить предметы на заданные места.

### Входные данные

В первой строке записано число  $n$  ( $1 \leq n \leq 50$ ), в следующих  $n$  строках содержатся описания предметов, в  $i$ -ой из которых, находится три числа координаты  $x_i, y_i$  ( $1 \leq x_i, y_i \leq 1000$ ) и максимальная скорость  $v_i$  ( $1 \leq v_i \leq 10$ )  $i$ -ого предмета соответственно. В следующих  $n$  строках содержатся описания финальных позиций, в  $i$ -ой из которых, заданы координаты  $a_i, b_i$  ( $1 \leq a_i, b_i \leq 1000$ )  $i$ -ой финальной позиции.

### Выходные данные

Выведите одно число — минимальное время, за которое можно переместить предметы. Погрешность не более  $10^{-4}$ .

### Примеры

<b>входные данные</b>	Copy
2 0 0 1 0 1 1 1 1 1 0	
<b>выходные данные</b>	Copy
1.0	

  

<b>входные данные</b>	Copy
2 0 0 1 0 1 1 1 1 2 1	
<b>выходные данные</b>	Copy
2.0	

  

<b>входные данные</b>	Copy
2 0 0 1 5 0 1 5 12 10 12	
<b>выходные данные</b>	Copy
13.0	



входные данные	Copy
2 0 0 2 5 0 1 5 12 10 12	
выходные данные	Copy
12.0	

входные данные	Copy
4 78 520 5 827 239 5 620 200 7 809 269 7 986 496 754 745 772 375 44 223	
выходные данные	Copy
68.45242650102003	

Statement is not available on English language

## G. Великая стена

ограничение по времени на тест: 2 секунды  
ограничение по памяти на тест: 256 мегабайт

**ввод: стандартный ввод**

**вывод: стандартный вывод**

У короля Людовика двое сыновей. Они ненавидят друг друга, и король боится, что после его смерти страна будет уничтожена страшными войнами. Поэтому Людовик решил разделить свою страну на две части, в каждой из которых будет властвовать один из его сыновей. Он

посадил их на трон в города  $A$  и  $B$ , и хочет построить минимально возможное количество фрагментов стены таким образом, чтобы не существовало пути из города  $A$  в город  $B$ .

Страну, в которой властвует Людовик, можно упрощенно представить в виде прямоугольника  $m \times n$ . В некоторых клетках этого прямоугольника расположены горы, по остальным же можно свободно перемещаться. Кроме этого, ландшафт в некоторых клетках удобен для строительства стены, в остальных же строительство невозможно.

При поездках по стране можно перемещаться из клетки в соседнюю по стороне, только если ни одна из этих клеток не содержит горы или построенного фрагмента стены.

### Входные данные

В первой строке входного файла содержатся числа  $m$  и  $n$  ( $1 \leq m, n \leq 50$ ). Следующие  $m$  строк по  $n$  символов задают карту страны. Символы обозначают: «#» — гора, «.» — место, пригодное для постройки стены, «-» — место, не пригодное для постройки стены, «A» и «B» — города  $A$  и  $B$ .

### Выходные данные

В первой строке выходного файла должно быть выведено минимальное количество фрагментов стены  $F$ , которые необходимо построить. Далее нужно вывести карту в том же формате, как во входном файле. Клетки со стеной обозначьте символом «+».

Если невозможно произвести требуемую застройку, то выведите в выходной файл единственное число - 1.

### Примеры

#### входные данные

[Copy](#)

```
5 5
--...
A- .#-
.##.-
--.-
--.-B
```

#### выходные данные

[Copy](#)

```
3
--+..
A-+#-
+##.-
--.-
--.-B
```

<b>ВХОДНЫЕ данные</b>	Copy
1 2 AB	
<b>ВЫХОДНЫЕ данные</b>	Copy
-1	

  

<b>ВХОДНЫЕ данные</b>	Copy
2 2 A# #B	
<b>ВЫХОДНЫЕ данные</b>	Copy
0 A# #B	

## H. Bring Them There

time limit per test: 2 seconds

memory limit per test: 256 megabytes

**input:** bring.in

**output:** bring.out

By the year 3141, the human civilization has spread all over the galaxy. The special hypertunnels are used to travel from one star system to another. To use the hypertunnel, you fly to a special location near the source star using your spaceship, activate the hyperjumper, fly through the hypertunnel, get out near your destination star and fly to the planet you need. The whole process takes exactly one day. A small drawback of the system is that for each tunnel every day only one spaceship can travel using this tunnel.

You are working in the transportation department of the "Intergalaxy Business Machines" company. This morning your boss has assigned a new task to you. To run the programming contest IBM needs to deliver  $K$  supercomputers from Earth where the company headquarters are located to the planet Eisiem. Since supercomputers are very large, one needs the whole spaceship to carry each supercomputer. You are asked to find a plan to deliver the supercomputers that takes as few days as possible. Since IBM is a very powerful corporation, you may assume that any time you need some tunnel for hyperjump, it is at your service. However, you still can use each tunnel only once a day.

## Input

The first line of the input file contains  $N$  — the number of star systems in the galaxy,  $M$  — the number of tunnels,  $K$  — the number of supercomputers to be delivered,  $S$  — the number of the solar system (the system where planet Earth is) and  $T$  — the number of the star system where planet Eisiem is ( $\{2 \leq N \leq 50\}$ ,  $\{1 \leq M \leq 200\}$ ,  $\{1 \leq K \leq 50\}$ ,  $\{1 \leq S, T \leq N\}$ ,  $\{S \neq T\}$ ).

Next  $M$  lines contain two different integer numbers each and describe tunnels. For each tunnel the numbers of star systems that it connects are given. The tunnel can be traveled in both directions, but remember that each day only one ship can travel through it, in particular, two ships cannot simultaneously travel through the same tunnel in opposite directions. No tunnel connects a star to itself and any two stars are connected by at most one tunnel.

## Output

On the first line of the output file print  $L$  — the fewest number of days needed to deliver  $K$  supercomputers from star system  $S$  to star system  $T$  using hypertunnels. Next  $L$  lines must describe the process. Each line must start with  $C_i$  — the number of ships that travel from one system to another this day.  $C_i$  pairs of integer numbers must follow, pair  $A, B$  means that the ship number  $A$  travels from its current star system to star system  $B$ .

It is guaranteed that there is a way to travel from star system  $S$  to star system  $T$ .

## Example

input

Copy

```
6 7 4 1 6
1 2
2 3
3 5
5 6
1 4
4 6
4 3
```

output

Copy

```
4
2 3 2 4 4
3 2 4 3 3 4 6
3 1 4 2 6 3 5
2 1 6 3 6
```

## I. Group tournament

time limit per test: 2 seconds  
memory limit per test: 256 megabytes  
**input: standard input**  
**output: standard output**

In our capitalist-run, dog-eat-dog world money is everything, and big sports are no exception. All participating teams have already bought enough points for the next season, and all the local hockey federation has to do now is distribute the results of the upcoming games. However, some teams felt generous and apart from buying points also bought the results of some matches. Initially the federation officials thought it would only make their life easier: the more games are fixed, the less work. It was only later that they understood their wrong and asked us to be a part of their scheme and help them distribute the results of the games in the upcoming season.

The local hockey tournament follows a round scheme:  $N$  teams participate, each team plays a game against each and every other team strictly once. Teams score points for games according to the following rules:

- If the winning team is defined at the end of the regulation time of the match, it scores 3 points, and the other team gets none.
- If a game is tied after regulation time, overtime ensues. In this case the winner gets 2 points and the loser gets 1. The overtime is unlimited and lasts until someone scores a goal.

Based on the tournament results, a team's score is calculated as the sum of its points earned in all games played.

### Input

The first line of the input file contains an integer  $N$  — the number of tournament participants ( $2 \leq N \leq 100$ ). Teams are numbered from 1 to  $N$ .

The following  $N$  lines of the file each contain  $N$  symbols and are in essence a tournament table for the given moment of time.

The symbol  $a_{ij}$  in the line  $i$  and position  $j$  denotes the result of a game scheduled to be played by team number  $i$  against team number  $j$  ( $1 \leq i, j \leq N$ ). It can be one of the following:

- 'W' — means that the team  $i$  will win a match against  $j$  in the regulation time
- 'w' — the team  $i$  will win in a match against  $j$  in the overtime
- 'L' — the team  $i$  will lose to the team  $j$  in the overtime
- 'l' — the team  $i$  will lose to the team  $j$  in the regulation time
- '.' — if the result of the game between  $i$  and  $j$  is not yet determined
- '#' — if  $i$  equals  $j$ , it means that there is no such game, i.e. a team cannot play against itself.

It is guaranteed that the table is correct. More formally:

- $a_{ij} = \text{'\#'} for all  $i = j$$
- if  $a_{ij} = \text{'.'}$ , then  $a_{ji} = \text{'.'}$
- $a_{ij} = \text{'\text{W}'} when and only when  $a_{ji} = \text{'\text{L}'}\mathbf{'}$$
- $a_{ij} = \text{'\text{w}'} when and only when  $a_{ji} = \text{'\text{l}'}\mathbf{'}$$

The last line of the input file contains  $N$  integers  $p_i$  — the number of points the  $i$ -th team must score ( $1 \leq i \leq N$ ).

## Output

The output file must contain a completely filled tournament table in the same format as that in the input file.

It is guaranteed that a solution exists. If there are several solutions, print any of them.

## Example

input

Copy

```
4
#..W
.#w.
.l#.
L..#
8 6 3 1
```

output

Copy

```
#www
l#wW
Ll#w
LLl#
```

Statement is not available on English language

## Ж. Тараканы

ограничение по времени на тест: 2 секунды  
ограничение по памяти на тест: 256 мегабайт

**ввод: стандартный ввод**

## вывод: стандартный вывод

Несколько чистоплотных тараканьих семей хотят поселиться в однокомнатной квартире. Квартира состоит из большой комнаты и кухни, соединённых узким коридором. Тараканов совершенно не интересует коридор сам по себе, однако они хотят иметь доступ и к кухне, и к комнате. Для того чтобы тараканья семья имела такой доступ, ей нужен индивидуальный транспортный путь через коридор. Тараканы — умные насекомые, поэтому они решили составить план коридора и определить, какое максимальное количество транспортных путей можно проложить через коридор.

В тараканьем плане коридор представляется бесконечной в обе стороны полосой шириной  $W$  сантиметров. Тараканы начертили прямоугольную систему координат, ось  $X$  которой параллельна направлению коридора. Хозяева квартиры расставили в коридоре некоторое количество массивных предметов. Каждый предмет является прямоугольником со сторонами, параллельными осям координат, и вершинами с целыми координатами в сантиметрах. Границы коридора задаются уравнениями  $y = 0$  и  $y = W$ . На координатной плоскости тараканы нарисовали квадратную сетку со стороной одной клетки, равной 1 см, начиная от границы коридора.

Тараканы договорились, что каждый транспортный путь является бесконечной в обе стороны цепью квадратных клеток. Цивилизованные тараканы никогда не прыгают, поэтому две подряд идущие клетки в этой цепи должны быть соседними по стороне. Цивилизованные тараканы всегда бегают по полу, поэтому никакая клетка цепи не может пересекаться с хозяйской вещью или выходить за пределы коридора. Тараканы не любят бегать по кругу, поэтому все клетки цепи различны. Тараканы не хотят давки в час пик, поэтому они требуют, чтобы различные транспортные пути не пересекались, то есть не имели общих клеток. Транспортный путь должен соединять кухню и комнату, поэтому разные концы цепи должны уходить в разные стороны коридора.

Ваша задача — определить максимальное количество транспортных путей, которые можно проложить через коридор.

### Входные данные

В первой строке входного файла записано два целых числа  $N$  и  $W$ .  $N$  — количество предметов в коридоре,  $W$  — ширина коридора в сантиметрах ( $0 \leq N \leq 5000$ ,  $0 < W \leq 10^9$ ).

Каждая из последующих  $N$  строк описывает одну из хозяйских вещей. Она содержит четыре целых числа  $X_1$ ,  $Y_1$ ,  $X_2$ ,  $Y_2$  — координаты двух противоположных углов прямоугольника ( $-10^9 \leq X_1 < X_2 \leq 10^9$ ,  $0 \leq Y_1 < Y_2 \leq W$ ). Предметы, расставленные в коридоре, могут пересекаться.

### Выходные данные

В выходной файл нужно вывести одно целое число — максимально возможное количество транспортных путей.

### Пример

входные данные

Copy

```
2 9
-4 4 -1 7
2 1 5 5
```

**выходные данные**

Copy

5

---

[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov  
The only programming contests Web 2.0 platform