

## Задача А. Паросочетание

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

Дан двудольный невзвешенный граф. Необходимо найти максимальное паросочетание.

### Формат входного файла

В первой строке входного файла три целых числа  $n$ ,  $m$  и  $k$  ( $1 \leq n, m \leq 200$ ,  $1 \leq k \leq n \times m$ ) — количество чисел в первой и второй долях, а также число ребер соответственно. Далее следуют  $k$  строк, в каждой из которых два числа  $a_i$  и  $b_i$ , что означает ребро между вершиной с номером  $a_i$  первой доли и вершиной с номером  $b_i$  второй доли. Вершины в обеих долях нумеруются с единицы.

### Формат выходного файла

В выходной файл выведите одно число — максимальное число ребер в паросочетании.

### Примеры

<code>matching.in</code>	<code>matching.out</code>
3 3 5 1 1 1 3 2 1 2 2 3 2	3

## Задача В. Максимальный поток

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

Задан ориентированный граф, каждое ребро которого обладает целочисленной пропускной способностью. Найдите максимальный поток из вершины с номером 1 в вершину с номером  $n$ .

### Формат входного файла

Первая строка входного файла содержит  $n$  и  $m$  — количество вершин и количество ребер графа ( $2 \leq n \leq 100$ ,  $1 \leq m \leq 1000$ ). Следующие  $m$  строк содержат по три числа: номера вершин, которые соединяет соответствующее ребро графа и его пропускную способность. Пропускные способности не превосходят  $10^5$ .

### Формат выходного файла

В выходной файл выведите одно число — величину максимального потока из вершины с номером 1 в вершину с номером  $n$ .

### Примеры

<code>maxflow.in</code>	<code>maxflow.out</code>
4 5 1 2 1 1 3 2 3 2 1 2 4 2 3 4 1	3

## Задача С. Минимальный разрез

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

Задан неориентированный граф, каждое ребро которого обладает целочисленной пропускной способностью. Вершина с номером 1 — исток. Вершина с номером  $n$  — сток. Требуется найти минимальный  $S - T$  разрез в этом графе.

Напомним, что  $S - T$  разрезом в графе называется пара дизъюнктивных множеств вершин  $S$  и  $T$ , таких что  $S \cup T = V$ ,  $s \in S$ ,  $t \in T$ . Мощностью разреза называется сумма пропускных способностей ребер, один из концов которого принадлежит  $S$ , а другой  $T$ .

### Формат входного файла

Первая строка входного файла содержит  $n$  и  $m$  — количество вершин и ребер графа соответственно ( $2 \leq n \leq 500$ ,  $1 \leq m \leq 10^4$ ). В следующих  $m$  строках содержатся по три числа: номера вершин  $u$  и  $v$ , которые соединяет ребро  $(u, v)$  и его пропускная способность. Пропускные способности не превосходят  $10^9$ .

### Формат выходного файла

В первой строке выходного файла выведите натуральное число  $k$  — количество вершин в множестве  $S$ . В следующей строке выведите  $k$  чисел, разделенных пробелом — номера вершин в множестве  $S$ .

### Примеры

<code>cut.in</code>	<code>cut.out</code>
4 4	2
1 2 2	1 2
2 4 1	
1 3 1	
3 4 2	

## Задача D. Покрытие путями

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

Задан ориентированный ациклических граф. Требуется определить минимальное количество непересекающихся путей, покрывающих все вершины.

### Формат входного файла

Первая строка входного файла содержит  $n$  и  $m$  — количество вершин и ребер графа соответственно ( $2 \leq n \leq 1000$ ,  $0 \leq m \leq 10^5$ ). В следующих  $m$  строках содержатся по два числа: номера вершин  $u$  и  $v$ , которые соединяет ребро  $(u, v)$ .

### Формат выходного файла

В первой строке выходного файла выведите натуральное число  $k$  — минимальное количество путей, необходимых, чтобы покрыть все вершины.

### Примеры

<code>paths.in</code>	<code>paths.out</code>
3 3 1 3 3 2 1 2	1

## Задача Е. Декомпозиция потока

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

Задан ориентированный граф, каждое ребро которого обладает целочисленной пропускной способностью. Найдите максимальный поток из вершины с номером 1 в вершину с номером  $n$  и постройте декомпозицию этого потока.

### Формат входного файла

Первая строка входного файла содержит  $n$  и  $m$  — количество вершин и количество ребер графа ( $2 \leq n \leq 500$ ,  $1 \leq m \leq 10000$ ). Следующие  $m$  строк содержат по три числа: номера вершин, которые соединяет соответствующее ребро графа и его пропускную способность. Пропускные способности не превосходят  $10^9$ .

### Формат выходного файла

В первую строку выходного файла выведите одно число — количество путей в декомпозиции максимального потока из вершины с номером 1 в вершину с номером  $n$ . Следующий строки должны содержать описания элементарных потоков, на который был разбит максимальный. Описание следует выводить в следующем формате: величина потока, количество ребер в пути, вдоль которого течет данный поток и номера ребер в этом пути. Ребра нумеруются с единицы в порядке появления во входном файле.

### Примеры

<code>decomposition.in</code>	<code>decomposition.out</code>
4 5	3
1 2 1	1 2 1 4
1 3 2	1 3 2 3 4
3 2 1	1 2 2 5
2 4 2	
3 4 1	

## Задача F. Циркуляция

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

Назовем *циркуляцией* поток величины 0. Дан ориентированный граф с нижними и верхними пропускными способностями, то есть для любых вершин  $i$  и  $j$  должно быть верно, что  $l_{ij} \leq f_{ij} \leq c_{ij}$ , где  $l_{ij}$  — нижняя граница, а  $c_{ij}$  — верхняя. Требуется найти циркуляцию в данном графе, удовлетворяющую данным ограничениям.

### Формат входного файла

В первой строке входного файла 2 целых числа  $N$  и  $M$  ( $1 \leq N \leq 200$ ,  $0 \leq M \leq 15000$ ). Далее следуют  $M$  строк, описывающие ребра графа. Каждая строка содержит 4 целых положительных числа  $i, j, l_{ij}$  и  $c_{ij}$  ( $0 \leq l_{ij} \leq c_{ij} \leq 10^5$ ), что означает, что ребро ведет из вершины с номером  $i$  в вершину с номером  $j$  с нижней границей  $l_{ij}$  и верхней  $c_{ij}$ . Гарантируется, что если в графе есть ребро из  $i$  в  $j$ , то нет ребра из  $j$  в  $i$ .

### Формат выходного файла

Если не существует циркуляции удовлетворяющей данным ограничения, выведите NO. Иначе на первой строке выведите YES. Далее в  $M$  строках должно содержаться по одному числу. В  $i$ -ой строке — величина потока по ребру на  $i$ -ой строке во входном файле. Напомним, что для любых  $i$  и  $j$  должно быть верно, что  $l_{ij} \leq f_{ij} \leq c_{ij}$ .

### Примеры

<code>circulation.in</code>	<code>circulation.out</code>
4 6 1 2 1 2 2 3 1 2 3 4 1 2 4 1 1 2 1 3 1 2 4 2 1 2	NO
4 6 1 2 1 3 2 3 1 3 3 4 1 3 4 1 1 3 1 3 1 3 4 2 1 3	YES 1 2 3 2 1 1

## Задача G. Задача о назначениях

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

Дана целочисленная матрица  $C$  размера  $n \times n$ . Требуется выбрать  $n$  ячеек так, чтобы в каждой строке и каждом столбце была выбрана ровно одна ячейка и сумма значений в выбранных ячейках было минимальна.

### Формат входного файла

Первая строка входного файла содержит  $n$  ( $2 \leq n \leq 300$ ). Каждая из последующих  $n$  строк содержит по  $n$  чисел:  $C_{ij}$ . Все значения во входном файле неотрицательны и не превосходят  $10^6$ .

### Формат выходного файла

В первую строку выходного файла выведите одно число — искомая минимизируемая величина. Далее выведите  $n$  строк по два числа в каждой — номер строки и столбца клетки, участвующей в оптимальном назначении.

Пары чисел можно выводить в произвольном порядке.

### Примеры

<code>assignment.in</code>	<code>assignment.out</code>
2	2
1 2	1 1
2 1	2 2

## Задача Н. Максимальный поток минимальной стоимости

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

Задан ориентированный граф, каждое ребро которого обладает пропускной способностью и стоимостью. Найдите максимальный поток минимальной стоимости из вершины с номером 1 в вершину с номером  $n$ .

### Формат входного файла

Первая строка входного файла содержит  $n$  и  $m$  — количество вершин и количество ребер графа ( $2 \leq n \leq 100$ ,  $1 \leq m \leq 1000$ ). Следующие  $m$  строк содержат по четыре целых числа: номера вершин, которые соединяет соответствующее ребро графа, его пропускную способность и его стоимость. Пропускные способности и стоимости не превосходят  $10^5$ .

### Формат выходного файла

В выходной файл выведите одно число — цену максимального потока минимальной стоимости из вершины с номером 1 в вершину с номером  $n$ . Ответ не превышает  $2^{63} - 1$ . Гарантируется, что в графе нет циклов отрицательной стоимости.

### Примеры

<code>mincost.in</code>	<code>mincost.out</code>
4 5 1 2 1 2 1 3 2 2 3 2 1 1 2 4 2 1 3 4 2 3	12

## Задача I. $k$ паросочетаний

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

Дан полный взвешенный двудольный граф с равным количеством вершин в долях. Требуется выбрать  $k$  максимальных попарно не пересекающихся паросочетаний так, чтобы их суммарный вес был минимален.

### Формат входного файла

Первая строка входного файла содержит  $n$  и  $k$  — количество вершин в каждой из долей и количество паросочетаний ( $2 \leq n \leq 50$ ,  $1 \leq k \leq n$ ). Каждая из последующих  $n$  строк содержит по  $n$  чисел:  $C_{ij}$  — вес ребра, ведущего из  $i$ -й вершины левой доли в  $j$ -ю правой.

Все значения во входном файле неотрицательны и не превосходят  $10^6$ .

### Формат выходного файла

В первую строку выходного файла выведите одно число — искомый суммарный вес паросочетаний. Следующие  $k$  строк должны содержать по  $n$  чисел — номера вершины, правой доли, соответствующие вершинам левой.

### Примеры

<code>multiassignment.in</code>	<code>multiassignment.out</code>
3 2	6
1 2 1	1 2 3
1 1 2	3 1 2
2 1 1	

## Задача J. Назначение на узкое место

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

Дан полный взвешенный двудольный граф с равным количеством вершин в долях. Требуется найти полное паросочетание, в котором минимальное ребро максимально.

### Формат входного файла

Первая строка входного файла содержит  $n$  и  $k$  — количество вершин в каждой из долей и количество паросочетаний ( $2 \leq n \leq 300$ ,  $1 \leq k \leq n$ ). Каждая из последующих  $n$  строк содержит по  $n$  чисел:  $C_{ij}$  — вес ребра, ведущего из  $i$ -й вершины левой доли в  $j$ -ю правой.

Все значения во входном файле неотрицательны и не превосходят  $10^6$ .

### Формат выходного файла

В первую строку выходного файла выведите одно число — вес минимального ребра в паросочетании.

### Примеры

<code>minimax.in</code>	<code>minimax.out</code>
2 1 2 2 1	2