XXXII НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА
Национален кръг
Хасково, 22 – 25 април 2016 г.
Група AB, 9 – 12 клас, Ден 1

Задача AB1. ПРЕНОМЕРИРАНЕ

Автор: Руско Шиков

Пред професор Оптимизоров е поставена сложна оптимизационна задача. В рамките на един проект трябва да бъдат изпълнени N задачи, номерирани с числата от 1 до N, като последователността на изпълнението им се определя от ориентиран, ацикличен граф. По-точно, номерата на задачите са разположени във върховете на ориентиран, ацикличен граф и, ако задачи с номера u и v са свързани с ребро, ориентирано от u към v, това означава, че задача с номер u трябва да бъде изпълнена преди задача с номер v (задачата с номер u ще наричаме **предшественик** на задачата с номер v, а задачата с номер v – **наследник** на задачата с номер u). От тук следва, че към изпълнението на дадена задача може да се пристъпи тогава, когато са изпълнени всичките ѝ предшественици. Няма да се спираме на същността на оптимизационната задача, нито на страхотния алгоритъм за решаването ѝ, измислен от професора, а само ще кажем, че този алгоритъм изисквал преномериране на върховете на графа (отново с числата от 1 до N) по много специален начин. Следва неговото обяснение.

Определение: Нека са дадени две числови редици $F = (f_1, f_2, \ldots, f_p)$ и $Q = (q_1, q_2, \ldots, q_r)$. Ако $p = 0$ или $r = 0$, то съответната редица е празна. Ще казваме, че редицата F лексикографски е по-малка от редицата Q, ако съществува такова i, че за всички $1 \le j < i$ е изпълнено $f_j = q_j$ и $f_i < q_i$ или, ако $p < r$ и $f_j = q_j$ за всяко $1 \le j \le p$. Ясно е, че лексикографски най-малка се явява празната редица.

Нека си представим, че след преномерирането на всеки връх на графа е съпоставена редица от новите номера на всичките му наследници, подредени в **намаляващ** ред. Такава редица да наречем **„наследствена“.** Новите номера трябва да бъдат такива, че един връх да има нов номер, по-малък от новия номер на друг връх, тогава и само тогава, когато наследствената му редица е лексикографски по-малка от наследствената редица на другия връх. Ако двата върха имат едно и също множество от наследници, по-малък нов номер трябва да има върхът с по-малък стар номер.

Помогнете на професора, като напишете програма **`renum`**, която извършва описаното преномериране на върховете на графа.

Вход

От първия ред на стандартния вход се въвеждат две цели положителни числа N и M, разделени с интервал – съответно брой на върховете и брой на ребрата в графа.

Следват M реда, всеки от които съдържа по две цели положителни числа, разделени с интервал – начален и краен връх на поредното ребро.

Изход

Вашата програма трябва да изведе N реда, на всеки от които има по две цели положителни числа, разделени с един интервал – първото число е старият номер на върха, а второто – новият. Редовете трябва да са сортирани в нарастващ ред по старите номера на върховете.

Бележка: Ако се понапънете малко, ще съобразите, че решението е единствено.

Ограничения

$2 \le N \le 100\,000$
$2 \le M \le 1\,000\,000$

В 20% от тестовете ($N \le 200$, $M \le 200$).

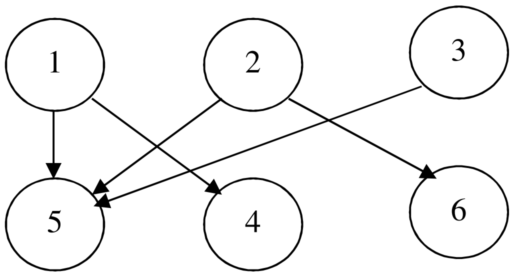
Пример

Вход	**Изход**

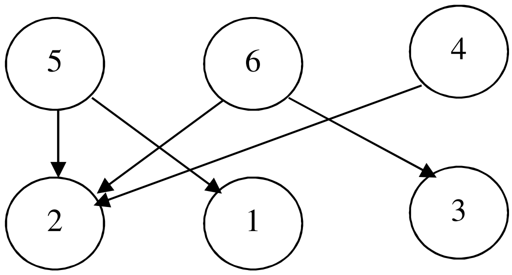
`6 5`
`1 5`
`2 5`
`3 5`
`1 4`
`2 6`
	`1 5`
	`2 6`
	`3 4`
	`4 1`
	`5 2`
	`6 3`

Обяснение на примера:

Граф преди преномерирането:



Граф след преномерирането:



Наследствените редици на върхове с нови номера 1, 2 и 3 са празни и, поради тази причина, те са получили първите три номера. Кой какъв номер да получи е било определено от стария му номер. Наследствените редици на върхове с нови номера 4, 5 и 6 са съответно {2}, {2, 1} и {3, 2}, което определя и получените от тях номера.