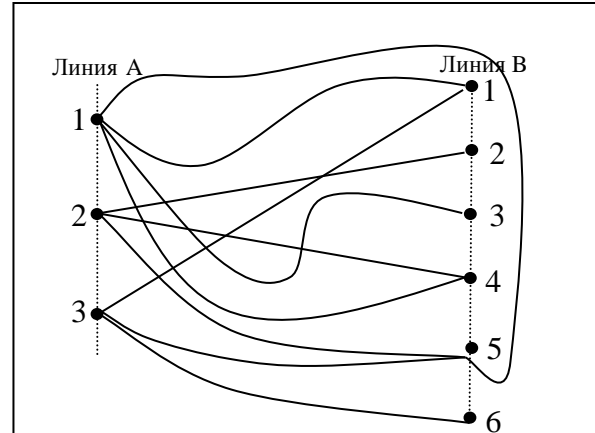


### Задача В3. ПЕЧАТНИ ПЛАТКИ

Изработването на печатни платки в електрониката е високотехнологичен и скъпо струващ процес: известен брой точки на специално изработена (евентуално многослойна) изолационна пластина (“платка”) се свързват чрез непресичащи се тънки електропроводими покрития (“пътечки”). На платките, с които разполагате, две мислени линии от точки А и В са съединени с пътечки, като някои точки от А са свързани с някои точки от В. Това много Ви устройва: електронната платка, която разработвате, трябва да има точно такава конструкция. Само че, във Вашия случай, всяка точка от А трябва да е свързана точно с две точки от В, а всяка точка от В – най-много с една точка от А. Дали можете да пригледите наличните платки за нуждите си и да реализирате сериозни икономии? Единственото оръжие, с което разполагате, е възможност да прекъснете някои от дадените пътечки – които искате и колкото искате: това е много по-евтино от създаването на нова платка. Напишете програмата **PLATE.EXE**, която да определя има ли задачата решение и кои от пътечките трябва да се прекъснат.



**ВХОД:** От стандартния вход се въвеждат:

- ред 1: трите естествени числа М (брой на точките на линия А), N (брой на точките на линия В) и К, разделени с интервал. М и N не надвишават 100, а К не е повече от 5000;
- на всеки от следващите К реда се описва една от пътечките чрез две естествени числа, разделени с интервал. Първото е номер на точка от линия А, а второто – от линия В.

**ИЗХОД:** Запишете на стандартния изход:

- един ред със съобщението NO, ако, уви, платката не може да бъде пригодена за нуждите Ви; иначе:
- ред 1: едно цяло неотрицателно число S – брой на пътечките, които ще бъдат прекъснати. Ако S е нула други редове няма;
- на всеки от следващите S реда опишете по една от пътечките, която ще бъде прекъсната в същия формат, в какъвто е дадена при въвеждането.

**ЗАБЕЛЕЖКА:** На схемата по-горе пътечките, свързващи точките, не се пресичат – те минават на различни слоеве.

#### ПРИМЕР

**Вход**

```
3 6 10
1 1
1 3
1 4
1 5
2 2
2 4
2 5
3 1
3 5
3 6
```

**Изход**

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