

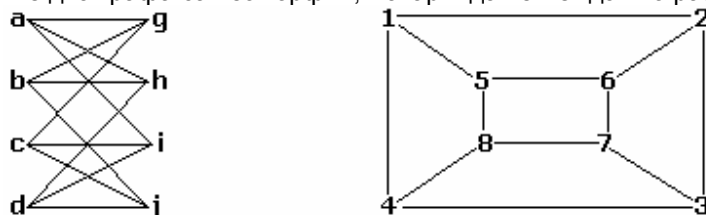


## ЗАБРАНЕН ПОДГРАФ

Два неориентирани графа  $G$  и  $H$  се наричат *изоморфни* когато:

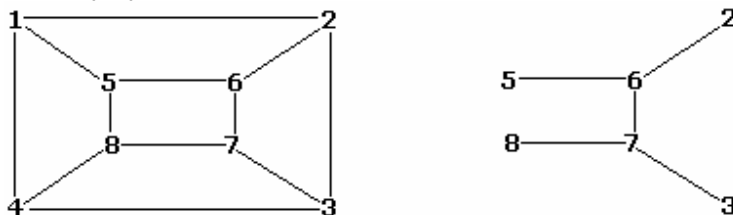
- имат един и същ брой върхове
- съществува биекция между върховете на единия и върховете на другия, така че между два върха от  $G$  има ребро, тогава и само тогава, когато има ребро между съответните върхове от графа  $H$ .

Например, следните два графа са изоморфни, макар и да изглеждат по различен начин:

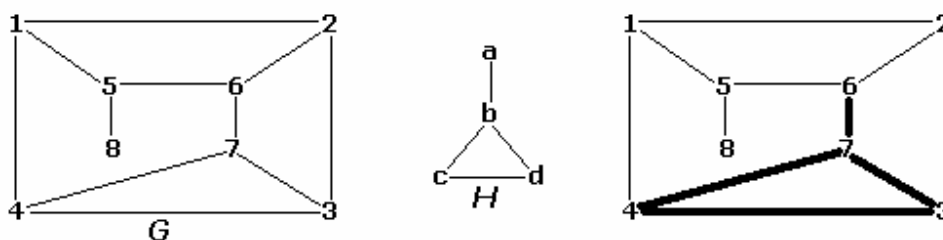


Една възможна биекция, показваща че двата графа са изоморфни е  $\{a-1, b-6, c-8, d-3, g-5, h-2, i-4, j-7\}$ , но не е единствената възможна.

*Подграф*  $G'$  на граф  $G$  наричаме граф с множество от върхове и множество от ребра, които са съответно подмножества на множеството от върховете и множеството на ребрата на графа  $G$ . Да отбележим, че всеки граф е подграф на себе си. Следният пример показва граф и един от неговите подграфи:



Казваме, че графът  $G$  *съдържа* друг граф  $H$ , ако съществува подграф  $H'$  на  $G$ , който е изоморфен на  $H$ . Фигурата показва граф  $G$ , който съдържа графа  $H$ .



## ЗАДАЧА

По дадени два неориентирани графа  $G$  и  $H$ , да се намери подграф  $G'$  на  $G$ , така че:

- броят на върховете на  $G$  и  $G'$  е един и същ
- $G'$  не съдържа  $H$ .

Естествено, може да има много подграфи  $G'$  с горните свойства. Да се изведе подграф, който има колкото е възможно повече ребра.



### БАЗОВ АЛГОРИТЪМ

Навярно една базова стратегия за решаване на проблема е следната: подреждаме ребрата на  $G$  в някакъв ред и се опитваме да ги добавяме едно по едно към  $G'$ , като на всяка стъпка проверяваме дали  $H$  се съдържа в  $G'$  или не. Тази стратегия, ако се реализира може да спечели някакви точки, но трябва да се отбележи, че съществуват много по-добри стратегии.

### ОГРАНИЧЕНИЯ

$3 \leq m \leq 4$  брой върхове на  $H$ .  
 $3 \leq n \leq 1000$  брой върхове на  $G$ .

### ВХОД

Дадени са 10 файла `forbidden1.in`, ..., `forbidden10.in` всеки със следните данни:

<code>forbiddenK.in</code>	ОПИСАНИЕ
<pre>3 5 0 1 0 1 0 1 0 1 0 0 1 0 0 0 1 0 1 0 0 0 1 0 1 0 0 0 1 0 1 0 0 0 1 0</pre>	<p><b>РЕД 1:</b> Contains two space-separated integers, respectively: <math>m</math> and <math>n</math>.</p> <p><b>СЛЕДВАЩИТЕ <math>m</math> РЕДА:</b> всеки ред съдържа <math>m</math> цели числа, разделени с интервали и представя един връх на графа <math>H</math> в ред 1, ..., <math>m</math>, като <math>i</math>-тия елемент на <math>j</math>-тия ред е равен на 1, когато върховете <math>i</math> и <math>j</math> са свързани с ребро в графа <math>H</math> и е 0 в противен случай.</p> <p><b>СЛЕДВАЩИТЕ <math>n</math> РЕДА:</b> всеки ред съдържа <math>n</math> цели числа, разделени с интервали и представя един връх на графа <math>G</math> в ред 1, ..., <math>n</math>, като <math>i</math>-тия елемент на <math>j</math>-тия ред е равен на 1, когато върховете <math>i</math> и <math>j</math> са свързани с ребро в графа <math>G</math> и е 0 в противен случай.</p>

Забележете, че като изключим ред 1, входът представлява матриците на съседство на  $H$  и  $G$ .

### ИЗХОД

Трябва да изпратите 10 файла по един за всеки от входните файлове. Всеки файл трябва да съдържа следните данни:

<code>forbiddenK.out</code>	ОПИСАНИЕ
<pre>#FILE forbidden K 5 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</pre>	<p><b>РЕД 1:</b> Заглавен ред, който трябва да съдържа <code>#FILE forbidden K</code> където <math>K</math> е число от 1 до 10, показващо за кой входен файл се отнася този изход.</p> <p><b>РЕД 2:</b> Съдържа едно число: <math>n</math>.</p> <p><b>СЛЕДВАЩИТЕ <math>n</math> РЕДА:</b> всеки ред съдържа <math>n</math> цели числа, разделени с интервали и представя един връх на графа <math>G'</math> в ред 1, ..., <math>n</math>, като <math>i</math>-тия елемент на <math>j</math>-тия ред е равен на 1, когато върховете <math>i</math> и <math>j</math> са свързани с ребро в графа <math>G'</math> и е 0 в противен случай.</p>

Забележете, че като изключим редове 1 и 2, изходът представлява матрицата на съседство на графа  $G'$ .



**ОЦЕНЯВАНЕ**

Точките, които ще получите зависят от броя на ребрата в графа  $G'$  от изхода. Определянето им става по следния начин: ще получите ненулев брой точки за всеки изход само, ако удовлетворява изискванията за задачата. Нека  $E_y$  е броят на ребрата във вашия изход,  $E_b$  е броят на ребрата в граф  $G'$ , получен с БАЗОВИЯ АЛГОРИТЪМ и  $E_m$  е максималният брой ребра в изхода на някой от състезателите. Точките за съответния тест се пресмятат по формулите:

- $30 E_y / E_b$ , ако  $E_y \leq E_b$  или
- $30 + 70(E_y - E_b) / (E_m - E_b)$ , ако  $E_y > E_b$ .