

Cvičenia č. 7, úloha č. 4

Podobne, ako sme v tretej úlohe hľadali cykly v orientovaných grafoch, budeme teraz hľadať *kružnice* v *neorientovaných* grafoch. *Kružnicou* v neorientovanom grafe nazveme uzavretý sled v_0, v_1, \dots, v_n taký, že $n \geq 1$ a vrcholy v_0, \dots, v_{n-1} , ako aj neorientované hrany $\{v_0, v_1\}, \{v_1, v_2\}, \dots, \{v_{n-1}, v_n\}$, sú po dvoch rôzne. Ide teda o sled nenulovej dĺžky, v ktorom sa počiatočný a koncový vrchol rovnajú, ale inak sa v ňom neopakujú žiadne vrcholy ani neorientované hrany. Podobne ako pri cykloch v orientovaných grafoch platí, že každá slučka je súčasne aj kružnicou.

Pozor: ak neorientovaný graf interpretujeme ako orientovaný tak, že neorientované hrany budeme chápať ako dvojice protichodných orientovaných hrán, *nebude* každý cyklus vo výslednom orientovanom grafe kružnicou v pôvodnom neorientovanom grafe: sled u, v, u pre vrcholy u a v spojené hranou netvorí kružnicu, pretože sa v ňom opakuje neorientovaná hrana $\{u, v\}$.

Priložený ZIP archív obsahuje triedy pre grafy z prednášky a tiež kostru triedy `UndirectedCycles`. Vašou úlohou je doplniť do tejto triedy implementáciu statickej metódy `belongsToCycle`, ktorá vezme ako argumenty neorientovaný graf `g` a jeho vrchol `vertex`; na výstupe táto metóda vráti booleovskú hodnotu podľa toho, či vrchol `vertex` leží na nejakej kružnici v grafe `g`. Výstupom tejto metódy teda má byť `true` práve vtedy, keď v grafe `g` existuje kružnica v_0, v_1, \dots, v_n taká, že niektorý z vrcholov v_0, v_1, \dots, v_n je rovný vrcholu `vertex`. Trieda `UndirectedCycles` už obsahuje hotovú implementáciu metódy `main`, ktorá načíta zo vstupu neorientovaný graf nasledovaný nejakým jeho vrcholom a pre túto dvojicu vstupov zavolá metódu `belongsToCycle`. V prípade potreby môžete do triedy `UndirectedCycles` doplniť aj ďalšie metódy. Implementáciu metódy `main` ale nemeňte.

Na testovač odovzdávajte iba súbor `UndirectedCycles.java` obsahujúci zdrojový kód vami upravenej triedy `UndirectedCycles` v balíku `graphs`. Zvyšné triedy balíka `graphs` k nej budú na testovači priložené.

Príklad vstupu č. 1:

```
4 3
0 1
1 2
2 3
0
```

Príklad výstupu č. 1:

```
Vrchol 0 nelezi na kruznici.
```

Príklad vstupu č. 2:

```
4 4
0 1
1 2
2 3
3 0
0
```

Príklad výstupu č. 2:

```
Vrchol 0 lezi na kruznici.
```