

Test č. 5, úloha č. 2

Pod *mostom* v neorientovanom grafe rozumieme takú jeho neorientovanú hranu, ktorej odstránením sa zvýši počet komponentov súvislosti grafu – jedinou cestou spájajúcou koncové vrcholy takejto hrany je teda táto hrana samotná. Všimnime si, že slučka nikdy nemôže byť mostom.

Priložený ZIP archív obsahuje balík `graphs` s triedami pre grafy z prednášky a s kostrou triedy `Bridges`, ktorá má realizovať hľadanie všetkých mostov v danom neorientovanom grafe. V triede `Bridges` doprogramujte:

- Konštruktor `public Bridges(UndirectedGraph g)`, ktorý nájde všetky mosty v neorientovanom grafe `g` a informáciu o nich si uloží do vhodnej premennej inštancie.
- Metódu `public int bridgeCount()`, ktorá vráti počet mostov nájdených konštruktorom.
- Metódu `public Set<UndirectedEdge> getBridges()`, ktorá vráti množinu neorientovaných hrán pozostávajúcu z práve všetkých mostov nájdených konštruktorom.

Pomocou výstupov metód inštancie triedy `Bridges` by nemalo byť možné modifikovať vnútorný stav tejto inštancie; nesplnenie tejto požiadavky môže mať za následok nesprávne výstupy na testovači.

Na testovač odovzdávajte iba súbor `Bridges.java` obsahujúci zdrojový kód vami doplnenej triedy. Pri tvorbe kódu rešpektujte konvencie jazyka Java a elementárne zásady objektovo orientovaného programovania.

Príklad. Graf na nasledujúcom obrázku obsahuje práve dva mosty, ktorými sú neorientované hrany $[0, 1]$ a $[4, 5]$.

