

Test č. 3, úloha č. 2

Orientovaný graf je *silno súvislý*, ak z každého vrchola vedie cesta do každého vrchola. V ľubovoľnom orientovanom grafe G je silno súvislý komponent C podgraf grafu G , ktorý je silno súvislý a maximálny (vzhľadom na inklúziu na množine vrcholov). Je to teda graf C , pre ktorý platí $V(C) \subseteq V(G)$, $E(C) \subseteq E(G)$, pre každé dva vrcholy $u, v \in V(C)$ existuje cesta z u do v a pre každý vrchol $v \in V(G) \setminus V(C)$ buď neexistuje cesta z v do vrcholov C alebo neexistuje cesta z vrcholov C do v . Inými slovami, ak na množine vrcholov grafu definujeme reláciu ekvivalencie $u \equiv v$ ak existuje cesta z u do v aj cesta z v do u , potom silno súvislé komponenty sú triedy ekvivalencie tejto relácie.

Stiahnite si kostru obsahujúcu implementáciu balíka `graphs` z prednášky a kostru triedy `StronglyConnectedComponents`. Vašou úlohou je doprogramovať implementáciu tejto triedy. Trieda `StronglyConnectedComponents` má obsahovať:

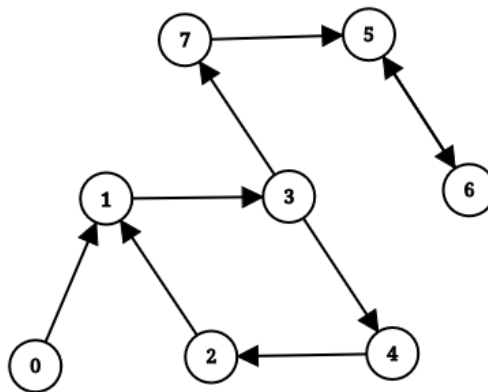
- Konštruktor, ktorý dostane orientovaný graf G .
- Metódu `public Set<Integer> getStronglyConnectedComponent(int vertex)`, ktorá dostane na vstupe vrchol grafu G a má vrátiť množinu vrcholov silno súvislého komponentu obsahujúceho vrchol `vertex`. Môžete predpokladať, že metóda bude volaná len s číslami zodpovedajúcimi vrcholom grafu G .

Na testovač odovzdajte súbor `StronglyConnectedComponents.java` obsahujúci vašu implementáciu tejto triedy. V kostre je aj trieda `TestSCC`, ktorá obsahuje funkciu na načítanie grafu. Môžete ju využiť na testovanie.

Nápoveda: Nájdenie silno súvislého komponentu obsahujúceho vrchol `vertex` sa dá urobiť tak, že sa nájde množina vrcholov dosiahnuteľná z vrchola `vertex`, z ktorej sa odstránia tie vrcholy, z ktorých nie je dosiahnuteľný vrchol `vertex`.

Príklad Nech je graf zadaný nasledovným vstupom, aj s ilustračným obrázkom :

```
9 9
0 1
1 3
3 4
4 2
2 1
3 7
7 5
5 6
6 5
```



Silno súvislé komponenty tohto grafu sú $\{0\}$, $\{1, 2, 3, 4\}$, $\{5, 6\}$ a $\{7\}$. Teda zavolanie metódy `getStronglyConnectedComponent` na čísle 0 vráti množinu $\{0\}$. Zavolanie na číslach 1, 2, 3 a 4 vráti množinu $\{1, 2, 3, 4\}$. Zavolanie na číslach 5 a 6 vráti $\{5, 6\}$ a Zavolanie na čísle 7 vráti $\{7\}$.