

Test č. 4, úloha č. 1

Vrcholovým pokrytím (angl. *vertex cover*) neorientovaného grafu G rozumieme množinu jeho vrcholov takú, že každá hrana grafu G má aspoň jeden z koncových vrcholov v tejto množine. Každá hrana je teda „pokrytá“ aspoň jedným vrcholom vrcholového pokrytia.

V tejto úlohe budeme navyše pracovať s grafmi, v ktorých sú jednotlivým vrcholom priradené celočíselné ceny. Cenou vrcholového pokrytia v takom prípade nazveme súčet cien prislúchajúcich k vrcholom, ktoré toto pokrytie tvoria. Cieľom úlohy bude nájsť niektoré spomedzi najlacnejších vrcholových pokrytí daného neorientovaného grafu. V špeciálnom prípade, keď sú všetky ceny vrcholov rovné jednej (alebo inému pevne danému kladnému celému číslu), tak pôjde o hľadanie najmenších vrcholových pokrytí.

Priložený archív obsahuje balík `graphs` s triedami z prednášky a kostru triedy `CheapestVertexCover`. Doprogramujte túto triedu tak, aby jej inštancie realizovali hľadanie ľubovoľného spomedzi *najlacnejších vrcholových pokrytí* grafu z argumentu konštruktora pri cenách vrcholov daných druhým argumentom konštruktora.

Trieda `CheapestVertexCover` by mala poskytovať:

- Konštruktor

```
public CheapestVertexCover(UndirectedGraph g, List<Integer> vertexPrices),
```

ktorý ako vstupy dostane neorientovaný graf g o n vrchoch a zoznam `vertexPrices` dĺžky n , ktorého prvky postupne reprezentujú ceny vrcholov $0, \dots, n-1$ grafu g . Pre tieto vstupy konštruktor nájde ľubovoľné spomedzi najlacnejších vrcholových pokrytí neorientovaného grafu g pri cenách vrcholov daných zoznamom `vertexPrices` a uloží si toto pokrytie do vhodnej premennej inštancie.

Môžete predpokladať korektnosť obidvoch vstupov konštruktora.

- Metódu

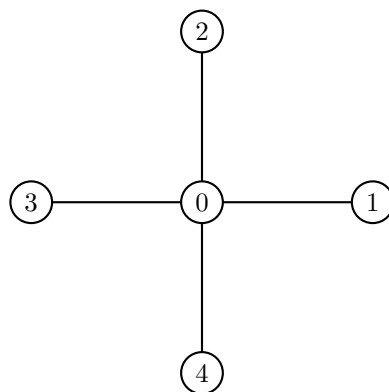
```
public Set<Integer> getCheapestVertexCover(),
```

ktorá na výstupe vráti nájdene najlacnejšie vrcholové pokrytie v podobe nemodifikovateľnej množiny vrcholov (obsahujúcej práve všetky vrcholy nájdeneho pokrytia).

V prípade potreby môžete do tejto triedy pridať aj ďalšie pomocné metódy.

Na testovač odovzdávajte iba súbor `CheapestVertexCover.java` obsahujúci kód vašej triedy.

Príklad 1. Uvažujme neorientovaný graf g na nasledujúcom obrázku.



Predpokladajme, že ceny jednotlivých vrcholov sú dané zoznamom $[5, 1, 1, 1, 1]$. Jediným najlacnejším vrcholovým pokrytím grafu g je potom množina obsahujúca vrcholy 1, 2, 3 a 4.

Príklad 2. Pokiaľ v rovnakom grafe ako vyššie zmeníme zoznam cien vrcholov na $[3, 1, 1, 1, 1]$, bude jeho jediným najlacnejším vrcholovým pokrytím množina obsahujúca iba vrchol 0.