

Test č. 2, úloha č. 2

Strom (*Tree*) je súvislý acyklický graf, t.j. súvislý graf, ktorý neobsahuje žiadnu kružnicu (cyklus). *Kostra* (*Spanning tree*) S grafu G je **podgraf** grafu G , ktorý je **strom** a obsahuje **všetky vrcholy** grafu G . Teda S je strom, pre ktorý platí $V(S) = V(G)$ a $E(S) \subseteq E(G)$.

Priložená kostra obsahuje triedy pre grafy z prednášky upravené nasledujúcim spôsobom. Implementácia orientovaných grafov bola odstránená, t.j. upravený balík `graphs` implementuje iba neorientované grafy. Trieda `UndirectedEdge` je upravená tak, že poskytuje metódy `getFrom` a `getTo`, rovnako ako poskytovala trieda `DirectedEdge`, namiesto metódy `getIncidentVertices`. Metóda `getFrom` vždy vracia vrchol s nižším indexom ako metóda `getTo`. Obe triedy pre neorientované grafy, t.j. `AdjacencyListsUndirectedGraph` a `AdjacencyMatrixUndirectedGraph` dedia od novej abstraktnej triedy `AbstractUndirectedGraph`, ktorá implementuje rozhranie `UndirectedGraph`.

Implementujte v abstraktnej triede `AbstractUndirectedGraph` nasledujúce metódy, ktoré boli doplnené do rozhrania `UndirectedGraph`.

- `public boolean isTree()`, ktorá vráti `true` práve vtedy, keď je daný graf strom.
- `public boolean isSpanningTree(Collection<? extends UndirectedEdge> edges)`, ktorá vráti `true` práve vtedy, keď hrany, ktoré dostane v kolekcii `edges`, tvoria kostru daného grafu.

Hint: Na zistenie či hrany v kolekcii `edges` tvoria strom, použite volanie metódy `isTree` na novom grafe, ktorý obsahuje práve tieto hrany.

Dodržiňte konvencie jazyka *Java* a základné princípy objektovo orientovaného programovania.

Na testovač odovzdávajte iba súbor `AbstractUndirectedGraph.java` obsahujúci vami doplnený kód abstraktnej triedy `AbstractUndirectedGraph`. Ostatné triedy balíka `graphs` budú k tejto triede na testovači priložené. Priložený archív tiež obsahuje triedu `Testovac`, ktorá obsahuje statické metódy na načítanie grafu a množiny hrán, okrem toho aj prázdnu metódou `main`. Môžete ju využiť na otestovanie vášho riešenia na nasledujúcich príkladoch.

Príklad č. 1:

```
graph.isTree()           // true
```

Vstup graf `graph`:

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

Príklad č. 2:

```
graph.isTree()           // false
graph.isSpanningTree(edges1) // true
graph.isSpanningTree(edges2) // false
```

Vstup graf `graph`:

```
4 6
0 1
1 2
2 3
3 0
0 2
1 3
```

Vstup kolekcia edges1

3
0 2
3 1
1 2

Vstup kolekcia edges2

4
0 1
1 2
2 3
3 0