

Zadanie č. 1 [10b]

Napište program, ktorý dokáže zdefinovať Petriho sieť na ktorej je možné spúšťať prechody. Použite objektovo-orientovaný prístup (dedenie a polymorfizmus), nezabudnite na správne použitie modifikátorov prístupu pre triedy a ich atribúty.

Podrobnejší popis zadania:

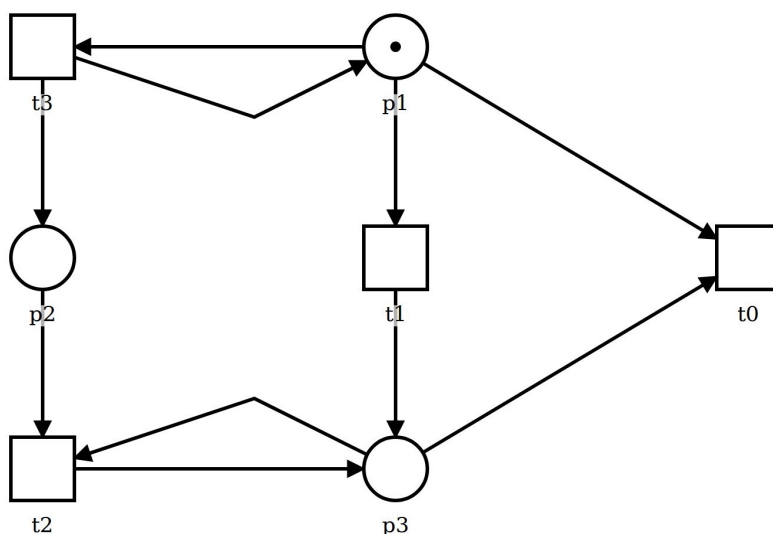
Petriho sieť je tvorená troma objektmi (Prechod, Miesto, Hrana). Petriho sieť dokáže spúšťať prechody a tým vie meniť značkovanie siete (počet značiek v miestach). Vytvorte si príslušné triedy, ktoré reprezentujú prechody, miesta, hrany a samotnú Petriho sieť. Trieda reprezentujúca Petriho sieť musí obsahovať metódu, ktorá spustí prechod identifikovaný svojim ID. Toto ID sa predáva ako argument do metódy. Petriho sieť musí podporovať hrany s násobnosťou viac ako 1. Hrany môžu byť: normálne alebo reset. Pri riešení zadania dbajte na rozšíriteľnosť, pretože budúce zadania budú nadväzovať.

V zadaní ošetríte nasledujúce prípady:

1. Ošetríte, aby sa dal spustiť iba spustiteľný prechod. *
2. Ošetríte, aby sa nedala vytvoriť hrana medzi vrcholmi rovnakými typu (place-place, transition-transition).*
3. Ošetríte, aby vytvorená hrana mala vždy nastavený vstupný a výstupný vrchol (aby sa nedala vytvoriť hrana, ktorá nikam nevedie a opačne). *
4. Ošetríte spustenie prechodu s hranou násobnosti viac ako 1.
5. Ošetríte, aby sa nedala vytvoriť hrana s násobnosťou menšou ako 1. *
6. Ošetríte, aby začiatočný bod reset hrany nemohol byť prechod. *

* treba použiť vhodnú výnimku pre daný prípad (nie všeobecnú výnimku Exception)

V main metóde vytvorte nový objekt Petriho siete reprezentujúci sieť na nasledujúcom obrázku:



Miesta: [p1, p2, p3]

Počiatočné značkovanie: [1, 0, 0]

Prechody: [t0, t1, t2, t3]

Hrany: [p1-t3, t3-p1, p1-t1, p1-t0, t3-p2, p2-t2, t1-p3, t2-p3, p3-t2, p3-t0]

Hodnotenie

Zadanie je hodnotené 10 bodmi. Odovzdaný program musí byť skompilovateľný, inak je hodnotený 0 bodmi. Hlavný dôraz v hodnotení sa kladie na objektový prístup a princípy, okrem iného:

- **vhodné pomenovanie** tried a metód v jednotnom jazyku (názvy tried s veľkým počiatočným písmenom, názvy metód s malým),
- vhodné použitie **modifikátorov prístupu** (public, private, poprípade protected) na obmedzenie prístupu k metódam a atribútom,
- využitie **dedenia a polymorfizmu**
- použitie **výnimiek** na ošetrovanie nedovoleného správania (nehádzať a nezachytávať všeobecnú triedu Exception),
- nepoužívajte [nested classy](#),
- vo vašich triedach nevytvárajte statické metódy (v zadani nie sú potrebné).
- V hlavnej triede nevytvárajte žiadnu logiku siete, iba vytvorte nový objekt vzorovej Petriho siete.

Odovzdávanie

Zadanie je potrebné odovzdať cez AIS do **24.3.2019 23:59**. Odovzdáva sa zozipovaný projekt (*.ZIP).