

## Cvičenia č. 3, úloha č. 1

Stiahnite si kostru k tejto úlohe, v ktorej nájdete všetky triedy pre uzly aritmetického stromu z tretej prednášky ako súčasť balíka `arithmetic`.

- a) Doprogramujte triedu `Times`, ktorej inštancie budú reprezentovať uzly aritmetického stromu pre binárnu operáciu násobenia. Podobne ako trieda `Plus`, mala by aj táto trieda dediť od abstraktnej triedy `BinaryNode`. Napíšte pre túto triedu konštruktor s dvoma parametrami – referenciami na ľavého a pravého syna daného uzla. Implementujte metódu `evaluate` pre túto triedu a prekryte tak abstraktnú metódu `evaluate` z triedy `Node`. Trieda `Times` by tiež mala byť súčasťou balíka `arithmetic`.
- b) Podstromy zakorenené v uzle typu `Times` by sme chceli prevádzať do textovej reprezentácie podobnej ako pri podstromoch zakorenených v uzloch typu `Plus`. Mohli by sme teda iba skopírovať metódu `toString` triedy `Plus` do triedy `Times` a zmeniť v reťazci, ktorý táto metóda vracia, znamienko `+` na `*`. Veľká časť konštrukcie reťazca by sa však potom v oboch triedach opakovala. Namiesto kopírovania metódy `toString` z triedy `Plus` do triedy `Times` teda *presuňte* túto metódu do triedy `BinaryNode` a znamienko v konštrukcii reťazca nahraďte volaním abstraktnej metódy `operatorToString` (bez parametrov a s návratovým typom `String`). Metódu `operatorToString` následne implementujte v triedach `Plus` a `Times` ako metódu, ktorá vracia reťazec `+` resp. `*`.

Na testovač odovzdávajte ZIP archív obsahujúci priečinok `arithmetic` a v ňom zdrojové súbory všetkých tried tohto balíka (vrátane tých, ktoré sa oproti kostre nezmenili). Testovač bude vytvárať inštancie tried z tohto balíka, volať ich metódy a bude tiež kontrolovať, či bola implementácia metódy `toString` presunutá do nadtriedy `BinaryNode`. Pri prekrývaní metód používajte anotáciu `@Override`.