Test č. 1, úloha č. 3

Pod *Laurentovým polynómom* s celočíselnými koeficientmi o jednej premennej x rozumieme konečný súčet tvaru

$$p(x) = a_m x^m + a_{m+1} x^{m+1} + \ldots + a_{n-1} x^{n-1} + a_n x^n,$$

kde $m, n \in \mathbb{Z}$ sú ľubovoľné *celé* čísla a $a_m, \ldots, a_n \in \mathbb{Z}$ sú celočíselné koeficienty. Laurentov polynóm sa teda od bežného polynómu odlišuje tým, že môže obsahovať aj členy s premennou x umocnenou na záporný exponent. Pre všetky $k \in \mathbb{Z}$ nazveme *koeficientom Laurentovho polynómu $p(x)$ pri x^k* celé číslo $[x^k]p(x)$ dané ako

$$[x^k]p(x) = \begin{cases} a_k & \text{ak } m \leq k \leq n, \\ 0 & \text{inak.} \end{cases}$$

Laurentov polynóm $p(x)$ nazveme *nulovým*, ak $[x^k]p(x) = 0$ pre všetky $k \in \mathbb{Z}$. *Stupňom* resp. *kostupňom* nenulového Laurentovho polynómu $p(x)$ nazveme najväčšie resp. najmenšie $k \in \mathbb{Z}$ také, že $[x^k]p(x) \neq 0$. *Stupeň* nulového Laurentovho polynómu definujeme ako $-\infty$ a *kostupeň* nulového Laurentovho polynómu ako ∞. *Súčtom* dvoch Laurentových polynómov $p_1(x)$ a $p_2(x)$ napokon nazveme Laurentov polynóm $p_1(x) + p_2(x)$ taký, že pre všetky $k \in \mathbb{Z}$ je $[x^k](p_1(x) + p_2(x)) = [x^k]p_1(x) + [x^k]p_2(x)$: koeficienty teda sčítame po zložkách, podobne ako pri bežných polynómoch.

Napíšte triedu `LaurentPolynomial` v nepomenovanom balíku reprezentujúcu (nemodifikovateľné) Laurentove polynómy s celočíselnými koeficientmi. Trieda by mala poskytovať konštruktor

```
public LaurentPolynomial(int minExponent, int[] coefficients)
```

ktorý pre hodnotu argumentu `minExponent` rovnú m a pole `coefficients` postupne obsahujúce celé čísla $a_0, \ldots, a_{n-1}$ vytvorí objekt reprezentujúci Laurentov polynóm

$$a_0 x^m + a_1 x^{m+1} + \ldots + a_{n-1} x^{m+n-1}.$$

Pole `coefficients` teda udáva niekoľko po sebe idúcich koeficientov Laurentovho polynómu počínajúc koeficientom pri x^m (exponent m tu môže byť aj záporný). Medzi nimi sú istotne všetky nenulové koeficienty vytváraného Laurentovho polynómu, avšak niektoré – a prípadne aj všetky – prvky poľa `coefficients` môžu byť aj nulové. V prípade, že je pole `coefficients` nulovej dĺžky, vytvorí sa konštruktorom reprezentácia nulového Laurentovho polynómu (a rovnako v prípade, keď pole `coefficients` obsahuje iba nulové prvky). Dajte si pozor na to, aby Laurentov polynóm vytvorený pomocou tohto konštruktora bol skutočne nemodifikovateľný.

Trieda `LaurentPolynomial` by ďalej mala poskytovať nasledujúce metódy:

- Metódu `public int getDegree()`, ktorá vráti *stupeň* reprezentovaného Laurentovho polynómu. V prípade nulového Laurentovho polynómu, ktorého stupeň definujeme ako $-\infty$, vráti táto metóda na výstupe číslo `Integer.MIN_VALUE`.
- Metódu `public int getCodegree()`, ktorá vráti *kostupeň* reprezentovaného Laurentovho polynómu. V prípade nulového Laurentovho polynómu, ktorého kostupeň definujeme ako ∞, vráti táto metóda na výstupe číslo `Integer.MAX_VALUE`.
- Metódu `public int getCoefficient(int n)`, ktorá vráti koeficient Laurentovho polynómu pri x^n. Argumentom tejto metódy môže byť ľubovoľné celé číslo n.
- Metódu `public LaurentPolynomial add(LaurentPolynomial p)`, ktorá sčíta Laurentov polynóm, pre ktorý je metóda volaná, s Laurentovým polynómom `p` z jej argumentu. Výstupom metódy je nová inštancia triedy `LaurentPolynomial` reprezentujúca súčet oboch Laurentových polynómov; volanie tejto metódy by nemalo nijakým spôsobom pozmeniť Laurentov polynóm, pre ktorý bola volaná, ani Laurentov polynóm `p` z jej argumentu.

Pri tvorbe kódu rešpektujte základné konvencie jazyka Java a metodické princípy objektovo orientovaného programovania. Na testovač odovzdávajte súbor `LaurentPolynomial.java` obsahujúci zdrojový kód triedy `LaurentPolynomial`. Testovač bude vykonávať rôzne postupnosti volaní konštruktora a metód tejto triedy.