

Test č. 1, úloha č. 1

Naprogramujte triedu `BigBinomialCoefficient` v nepomenovanom balíku, ktorá bude reprezentovať a vyhodnocovať (aj veľmi veľké) binomické koeficienty

$$\binom{n}{k} = \frac{n^k}{k!} = \frac{\prod_{j=0}^{k-1} (n-j)}{k!}.$$

Trieda `BigBinomialCoefficient` by mala poskytovať:

- Konštruktor

```
public BigBinomialCoefficient(int n, int k),
```

ktorý vytvorí objekt reprezentujúci binomický koeficient $\binom{n}{k}$. Môžete predpokladať, že n aj k sú nezáporné celé čísla, avšak n môže byť aj menšie ako k .

- Metódu `public int getUpperIndex()`, ktorá pre binomický koeficient $\binom{n}{k}$ vráti hodnotu čísla n .
- Metódu `public int getLowerIndex()`, ktorá pre binomický koeficient $\binom{n}{k}$ vráti hodnotu čísla k .
- Metódu `public BigInteger evaluate()`, ktorá vráti hodnotu reprezentovaného binomického koeficientu v podobe inštancie triedy `BigInteger` z balíka `java.math`.

Testovač bude vytvárať inštancie vašej triedy `BigBinomialCoefficient` a volať ich metódy. Príklad takejto postupnosti volaní aj s očakávanými výstupmi možno nájsť nižšie.

Okrem správnosti sa budú hodnotiť aj niektoré ďalšie aspekty odovzdaného riešenia – predovšetkým dodržiavanie elementárnych zásad objektovo orientovaného programovania (hlavne princípu zapuzdrenia) a rešpektovanie konvencií jazyka Java.

Na testovač odovzdávajte súbor `BigBinomialCoefficient.java` obsahujúci zdrojový kód vašej triedy.

Príklad. Predpokladajme, že inštancia triedy `BigBinomialCoefficient` bola vytvorená pomocou volania

```
BigBinomialCoefficient coef = new BigBinomialCoefficient(200, 100);
```

ide teda o reprezentáciu binomického koeficientu $\binom{200}{100}$.

Volania `coef.getUpperIndex()` a `coef.getLowerIndex()` potom postupne vrátia celé čísla 200 a 100. Následné volanie `coef.evaluate()` vráti inštanciu triedy `BigInteger` reprezentujúcu celé číslo

90548514656103281165404177077484163874504589675413336841320.