**Вбудована арифметика системи Java.**

**1. Основні відомості**

Арифметика багатократної точності представлена у мові Java класом BigInteger, який знаходиться у пакеті java.math. Даний клас реалізує знакову арифметику із може працювати із числами довжиною до біт включно. BigInteger має досить типову серед бібліотек багатократної точності реалізацію, основними полями об’єкту даного класу є наступні:

1. int[] mag – масив, що зберігає розряди числа у порядку big-endian, тобто старший байт даного числа містить у елементі mag[0]. Оскільки тип int мови Java, незалежно від платформи на якій виконується код, займає 4 байти, то кожен розряд відповідного довгого числа містить 32 біти.
2. int signum – поле, яке зберігає знак числа. Додатнім числам відповідає 1, від’ємним -1, а значення 0 свідчить про те, що значення відповідного числа теж дорівнює нулю.

Важливою особливістю реалізації класу BigInteger є те, що об’єкти даного класу є незмінюваними (immutable). Це означає, шо маючи об’єкт класу BigInteger ви більше не можете змінити його внутрішній стан, а всі методи, які виконують арифметичні та інші операції будуть повертати новий об’єкт даного класу, який представляє результат виконаних обчислень.

Іншою особливістю, яка характерна не тільки для класу BigInteger, а для мови Java в цілому, є використання виключень (exceptions) замість кодів повернення. Це означає, що якщо деяку операцію виконано успішно, то ми отримаємо її результат. Якщо ні, то буде викинуто виключення деякого класу. Які саме виключення у яких ситуаціях можна отримати детально описано у документації на офіційному сайті.

**2. Основні методи класу BigInteger**

Далі наведено перелік основних методів класу BigInteger, згрупованих за їх призначенням.

**2.1 Створення нового екземпляру класу BigInteger**

Створити новий об’єкт класу BigInteger можна за допомогою конструкторів та деяких службових статичних методів.

**BigInteger(byte[] val)**

Перетворює байтовий масив, що зберігає двійкове представлення об’єкту класу BigInteger у новий об’єкт даного класу. Мається на увазі, що байти розташовані у масиві у порядку big-endian.

**Параметри:**

byte[] val–масив, що містить двійкове представлення об’єкта класу BigInteger.

**Виключення:**

NumberFormatException – якщо масив val має нульову довжину.

**BigInteger(String val, int radix)**

Перетворює строкове представлення числа у заданій системі числення у об’єкт класу BigInteger.

Якщо другий параметр не вказано, то буде використано значення за замовчуванням – 10.

**Параметри:**

String val – рядок, що містить строкове представлення числа.

int radix – основа системи числення, яку треба використати при інтерпретації val.

**Виключення:**

NumberFormatException – якщо val не є коректним представленням числа у системі числення із основою radix, або якщо radix виходить за межі підтримуваного діапазону, тобто менше 2 або більше 36.

**static BigInteger probablePrime(int bitLength, Random rnd)**

Повертає об’єкт класу BigInteger, який є ймовірно простим числом, заданої довжини. Ймовірність того, що отримане число буде складним не перевищує .

**Параметри:**

int bitLength – довжина числа, яке буде повернуто

Random rnd – ГПСЧ, що використовується для геренації кандидатів для подальшого тестування на простоту

**Повертає:**

Новий об’єкт класу BigInteger, що представляє ймовірно просте число заданої довжини.

**Виключення:**

ArithmeticException – якщо bitLength < 2 або якщо bitLength перевищує діапазон підтримуваних значень.

**static BigInteger valueOf(long value)**

Повертає об’єкт класу BigInteger, значення якого відповідає переданому параметру.

**Параметри:**

long value– число, що відповідає об’єкту BigInteger який буде повернено

**Повертає:**

Новий об’єкт класу BigInteger, значення якого відповідає переданому параметру.