# Changelog

All notable changes to this project will be documented in this file.

## [0.12.1](https://github.com/brick/math/releases/tag/0.12.1) - 2023-11-29

⚡️ **Performance improvements**

* BigNumber::of() is now faster, thanks to [@SebastienDug](https://github.com/SebastienDug) in [#77](https://github.com/brick/math/pull/77).

## [0.12.0](https://github.com/brick/math/releases/tag/0.12.0) - 2023-11-26

💥 **Breaking changes**

* Minimum PHP version is now 8.1
* RoundingMode is now an enum; if you're type-hinting rounding modes, you need to type-hint against RoundingMode instead of int now
* BigNumber classes do not implement the Serializable interface anymore (they use the [new custom object serialization mechanism](https://wiki.php.net/rfc/custom_object_serialization))
* The following breaking changes only affect you if you're creating your own BigNumber subclasses:
  + the return type of BigNumber::of() is now static
  + BigNumber has a new abstract method from()
  + all public and protected functions of BigNumber are now final

## [0.11.0](https://github.com/brick/math/releases/tag/0.11.0) - 2023-01-16

💥 **Breaking changes**

* Minimum PHP version is now 8.0
* Methods accepting a union of types are now strongly typed\*
* MathException now extends Exception instead of RuntimeException

\* You may now run into type errors if you were passing `Stringable` objects to `of()` or any of the methods internally calling `of()`, with `strict\_types` enabled. You can fix this by casting `Stringable` objects to `string` first.

## [0.10.2](https://github.com/brick/math/releases/tag/0.10.2) - 2022-08-11

👌 **Improvements**

* BigRational::toFloat() now simplifies the fraction before performing division (#73) thanks to @olsavmic

## [0.10.1](https://github.com/brick/math/releases/tag/0.10.1) - 2022-08-02

✨ **New features**

* BigInteger::gcdMultiple() returns the GCD of multiple BigInteger numbers

## [0.10.0](https://github.com/brick/math/releases/tag/0.10.0) - 2022-06-18

💥 **Breaking changes**

* Minimum PHP version is now 7.4

## [0.9.3](https://github.com/brick/math/releases/tag/0.9.3) - 2021-08-15

🚀 **Compatibility with PHP 8.1**

* Support for custom object serialization; this removes a warning on PHP 8.1 due to the Serializable interface being deprecated (#60) thanks @TRowbotham

## [0.9.2](https://github.com/brick/math/releases/tag/0.9.2) - 2021-01-20

🐛 **Bug fix**

* Incorrect results could be returned when using the BCMath calculator, with a default scale set with bcscale(), on PHP >= 7.2 (#55).

## [0.9.1](https://github.com/brick/math/releases/tag/0.9.1) - 2020-08-19

✨ **New features**

* BigInteger::not() returns the bitwise NOT value

🐛 **Bug fixes**

* BigInteger::toBytes() could return an incorrect binary representation for some numbers
* The bitwise operations and(), or(), xor() on BigInteger could return an incorrect result when the GMP extension is not available

## [0.9.0](https://github.com/brick/math/releases/tag/0.9.0) - 2020-08-18

👌 **Improvements**

* BigNumber::of() now accepts .123 and 123. formats, both of which return a BigDecimal

💥 **Breaking changes**

* Deprecated method BigInteger::powerMod() has been removed - use modPow() instead
* Deprecated method BigInteger::parse() has been removed - use fromBase() instead

## [0.8.17](https://github.com/brick/math/releases/tag/0.8.17) - 2020-08-19

🐛 **Bug fix**

* BigInteger::toBytes() could return an incorrect binary representation for some numbers
* The bitwise operations and(), or(), xor() on BigInteger could return an incorrect result when the GMP extension is not available

## [0.8.16](https://github.com/brick/math/releases/tag/0.8.16) - 2020-08-18

🚑 **Critical fix**

* This version reintroduces the deprecated BigInteger::parse() method, that has been removed by mistake in version 0.8.9 and should have lasted for the whole 0.8 release cycle.

✨ **New features**

* BigInteger::modInverse() calculates a modular multiplicative inverse
* BigInteger::fromBytes() creates a BigInteger from a byte string
* BigInteger::toBytes() converts a BigInteger to a byte string
* BigInteger::randomBits() creates a pseudo-random BigInteger of a given bit length
* BigInteger::randomRange() creates a pseudo-random BigInteger between two bounds

💩 **Deprecations**

* BigInteger::powerMod() is now deprecated in favour of modPow()

## [0.8.15](https://github.com/brick/math/releases/tag/0.8.15) - 2020-04-15

🐛 **Fixes**

* added missing ext-json requirement, due to BigNumber implementing JsonSerializable

⚡️ **Optimizations**

* additional optimization in BigInteger::remainder()

## [0.8.14](https://github.com/brick/math/releases/tag/0.8.14) - 2020-02-18

✨ **New features**

* BigInteger::getLowestSetBit() returns the index of the rightmost one bit

## [0.8.13](https://github.com/brick/math/releases/tag/0.8.13) - 2020-02-16

✨ **New features**

* BigInteger::isEven() tests whether the number is even
* BigInteger::isOdd() tests whether the number is odd
* BigInteger::testBit() tests if a bit is set
* BigInteger::getBitLength() returns the number of bits in the minimal representation of the number

## [0.8.12](https://github.com/brick/math/releases/tag/0.8.12) - 2020-02-03

🛠️ **Maintenance release**

Classes are now annotated for better static analysis with [psalm](https://psalm.dev/).

This is a maintenance release: no bug fixes, no new features, no breaking changes.

## [0.8.11](https://github.com/brick/math/releases/tag/0.8.11) - 2020-01-23

✨ **New feature**

BigInteger::powerMod() performs a power-with-modulo operation. Useful for crypto.

## [0.8.10](https://github.com/brick/math/releases/tag/0.8.10) - 2020-01-21

✨ **New feature**

BigInteger::mod() returns the **modulo** of two numbers. The *modulo* differs from the *remainder* when the signs of the operands are different.

## [0.8.9](https://github.com/brick/math/releases/tag/0.8.9) - 2020-01-08

⚡️ **Performance improvements**

A few additional optimizations in BigInteger and BigDecimal when one of the operands can be returned as is. Thanks to @tomtomsen in #24.

## [0.8.8](https://github.com/brick/math/releases/tag/0.8.8) - 2019-04-25

🐛 **Bug fixes**

* BigInteger::toBase() could return an empty string for zero values (BCMath & Native calculators only, GMP calculator unaffected)

✨ **New features**

* BigInteger::toArbitraryBase() converts a number to an arbitrary base, using a custom alphabet
* BigInteger::fromArbitraryBase() converts a string in an arbitrary base, using a custom alphabet, back to a number

These methods can be used as the foundation to convert strings between different bases/alphabets, using BigInteger as an intermediate representation.

💩 **Deprecations**

* BigInteger::parse() is now deprecated in favour of fromBase()

BigInteger::fromBase() works the same way as parse(), with 2 minor differences:

* the $base parameter is required, it does not default to 10
* it throws a NumberFormatException instead of an InvalidArgumentException when the number is malformed

## [0.8.7](https://github.com/brick/math/releases/tag/0.8.7) - 2019-04-20

**Improvements**

* Safer conversion from float when using custom locales
* **Much faster** NativeCalculator implementation 🚀

You can expect **at least a 3x performance improvement** for common arithmetic operations when using the library on systems without GMP or BCMath; it gets exponentially faster on multiplications with a high number of digits. This is due to calculations now being performed on whole blocks of digits (the block size depending on the platform, 32-bit or 64-bit) instead of digit-by-digit as before.

## [0.8.6](https://github.com/brick/math/releases/tag/0.8.6) - 2019-04-11

**New method**

BigNumber::sum() returns the sum of one or more numbers.

## [0.8.5](https://github.com/brick/math/releases/tag/0.8.5) - 2019-02-12

**Bug fix**: of() factory methods could fail when passing a float in environments using a LC\_NUMERIC locale with a decimal separator other than '.' (#20).

Thanks @manowark 👍

## [0.8.4](https://github.com/brick/math/releases/tag/0.8.4) - 2018-12-07

**New method**

BigDecimal::sqrt() calculates the square root of a decimal number, to a given scale.

## [0.8.3](https://github.com/brick/math/releases/tag/0.8.3) - 2018-12-06

**New method**

BigInteger::sqrt() calculates the square root of a number (thanks @peter279k).

**New exception**

NegativeNumberException is thrown when calling sqrt() on a negative number.

## [0.8.2](https://github.com/brick/math/releases/tag/0.8.2) - 2018-11-08

**Performance update**

* Further improvement of toInt() performance
* NativeCalculator can now perform some multiplications more efficiently

## [0.8.1](https://github.com/brick/math/releases/tag/0.8.1) - 2018-11-07

Performance optimization of toInt() methods.

## [0.8.0](https://github.com/brick/math/releases/tag/0.8.0) - 2018-10-13

**Breaking changes**

The following deprecated methods have been removed. Use the new method name instead:

| **Method removed** | **Replacement method** |
| --- | --- |
| BigDecimal::getIntegral() | BigDecimal::getIntegralPart() |
| BigDecimal::getFraction() | BigDecimal::getFractionalPart() |

**New features**

BigInteger has been augmented with 5 new methods for bitwise operations:

| **New method** | **Description** |
| --- | --- |
| and() | performs a bitwise AND operation on two numbers |
| or() | performs a bitwise OR operation on two numbers |
| xor() | performs a bitwise XOR operation on two numbers |
| shiftedLeft() | returns the number shifted left by a number of bits |
| shiftedRight() | returns the number shifted right by a number of bits |

Thanks to @DASPRiD 👍

## [0.7.3](https://github.com/brick/math/releases/tag/0.7.3) - 2018-08-20

**New method:** BigDecimal::hasNonZeroFractionalPart()

**Renamed/deprecated methods:**

* BigDecimal::getIntegral() has been renamed to getIntegralPart() and is now deprecated
* BigDecimal::getFraction() has been renamed to getFractionalPart() and is now deprecated

## [0.7.2](https://github.com/brick/math/releases/tag/0.7.2) - 2018-07-21

**Performance update**

BigInteger::parse() and toBase() now use GMP's built-in base conversion features when available.

## [0.7.1](https://github.com/brick/math/releases/tag/0.7.1) - 2018-03-01

This is a maintenance release, no code has been changed.

* When installed with --no-dev, the autoloader does not autoload tests anymore
* Tests and other files unnecessary for production are excluded from the dist package

This will help make installations more compact.

## [0.7.0](https://github.com/brick/math/releases/tag/0.7.0) - 2017-10-02

Methods renamed:

* BigNumber:sign() has been renamed to getSign()
* BigDecimal::unscaledValue() has been renamed to getUnscaledValue()
* BigDecimal::scale() has been renamed to getScale()
* BigDecimal::integral() has been renamed to getIntegral()
* BigDecimal::fraction() has been renamed to getFraction()
* BigRational::numerator() has been renamed to getNumerator()
* BigRational::denominator() has been renamed to getDenominator()

Classes renamed:

* ArithmeticException has been renamed to MathException

## [0.6.2](https://github.com/brick/math/releases/tag/0.6.2) - 2017-10-02

The base class for all exceptions is now MathException. ArithmeticException has been deprecated, and will be removed in 0.7.0.

## [0.6.1](https://github.com/brick/math/releases/tag/0.6.1) - 2017-10-02

A number of methods have been renamed:

* BigNumber:sign() is deprecated; use getSign() instead
* BigDecimal::unscaledValue() is deprecated; use getUnscaledValue() instead
* BigDecimal::scale() is deprecated; use getScale() instead
* BigDecimal::integral() is deprecated; use getIntegral() instead
* BigDecimal::fraction() is deprecated; use getFraction() instead
* BigRational::numerator() is deprecated; use getNumerator() instead
* BigRational::denominator() is deprecated; use getDenominator() instead

The old methods will be removed in version 0.7.0.

## [0.6.0](https://github.com/brick/math/releases/tag/0.6.0) - 2017-08-25

* Minimum PHP version is now [7.1](https://gophp71.org/); for PHP 5.6 and PHP 7.0 support, use version 0.5
* Deprecated method BigDecimal::withScale() has been removed; use toScale() instead
* Method BigNumber::toInteger() has been renamed to toInt()

## [0.5.4](https://github.com/brick/math/releases/tag/0.5.4) - 2016-10-17

BigNumber classes now implement [JsonSerializable](http://php.net/manual/en/class.jsonserializable.php). The JSON output is always a string.

## [0.5.3](https://github.com/brick/math/releases/tag/0.5.3) - 2016-03-31

This is a bugfix release. Dividing by a negative power of 1 with the same scale as the dividend could trigger an incorrect optimization which resulted in a wrong result. See #6.

## [0.5.2](https://github.com/brick/math/releases/tag/0.5.2) - 2015-08-06

The $scale parameter of BigDecimal::dividedBy() is now optional again.

## [0.5.1](https://github.com/brick/math/releases/tag/0.5.1) - 2015-07-05

**New method: BigNumber::toScale()**

This allows to convert any BigNumber to a BigDecimal with a given scale, using rounding if necessary.

## [0.5.0](https://github.com/brick/math/releases/tag/0.5.0) - 2015-07-04

**New features**

* Common BigNumber interface for all classes, with the following methods:
  + sign() and derived methods (isZero(), isPositive(), ...)
  + compareTo() and derived methods (isEqualTo(), isGreaterThan(), ...) that work across different BigNumber types
  + toBigInteger(), toBigDecimal(), toBigRational() conversion methods
  + toInteger() and toFloat() conversion methods to native types
* Unified of() behaviour: every class now accepts any type of number, provided that it can be safely converted to the current type
* New method: BigDecimal::exactlyDividedBy(); this method automatically computes the scale of the result, provided that the division yields a finite number of digits
* New methods: BigRational::quotient() and remainder()
* Fine-grained exceptions: DivisionByZeroException, RoundingNecessaryException, NumberFormatException
* Factory methods zero(), one() and ten() available in all classes
* Rounding mode reintroduced in BigInteger::dividedBy()

This release also comes with many performance improvements.

**Breaking changes**

* BigInteger:
  + getSign() is renamed to sign()
  + toString() is renamed to toBase()
  + BigInteger::dividedBy() now throws an exception by default if the remainder is not zero; use quotient() to get the previous behaviour
* BigDecimal:
  + getSign() is renamed to sign()
  + getUnscaledValue() is renamed to unscaledValue()
  + getScale() is renamed to scale()
  + getIntegral() is renamed to integral()
  + getFraction() is renamed to fraction()
  + divideAndRemainder() is renamed to quotientAndRemainder()
  + dividedBy() now takes a **mandatory** $scale parameter **before** the rounding mode
  + toBigInteger() does not accept a $roundingMode parameter anymore
  + toBigRational() does not simplify the fraction anymore; explicitly add ->simplified() to get the previous behaviour
* BigRational:
  + getSign() is renamed to sign()
  + getNumerator() is renamed to numerator()
  + getDenominator() is renamed to denominator()
  + of() is renamed to nd(), while parse() is renamed to of()
* Miscellaneous:
  + ArithmeticException is moved to an Exception\ sub-namespace
  + of() factory methods now throw NumberFormatException instead of InvalidArgumentException

## [0.4.3](https://github.com/brick/math/releases/tag/0.4.3) - 2016-03-31

Backport of two bug fixes from the 0.5 branch:

* BigInteger::parse() did not always throw InvalidArgumentException as expected
* Dividing by a negative power of 1 with the same scale as the dividend could trigger an incorrect optimization which resulted in a wrong result. See #6.

## [0.4.2](https://github.com/brick/math/releases/tag/0.4.2) - 2015-06-16

New method: BigDecimal::stripTrailingZeros()

## [0.4.1](https://github.com/brick/math/releases/tag/0.4.1) - 2015-06-12

Introducing a BigRational class, to perform calculations on fractions of any size.

## [0.4.0](https://github.com/brick/math/releases/tag/0.4.0) - 2015-06-12

Rounding modes have been removed from BigInteger, and are now a concept specific to BigDecimal.

BigInteger::dividedBy() now always returns the quotient of the division.

## [0.3.5](https://github.com/brick/math/releases/tag/0.3.5) - 2016-03-31

Backport of two bug fixes from the 0.5 branch:

* BigInteger::parse() did not always throw InvalidArgumentException as expected
* Dividing by a negative power of 1 with the same scale as the dividend could trigger an incorrect optimization which resulted in a wrong result. See #6.

## [0.3.4](https://github.com/brick/math/releases/tag/0.3.4) - 2015-06-11

New methods:

* BigInteger::remainder() returns the remainder of a division only
* BigInteger::gcd() returns the greatest common divisor of two numbers

## [0.3.3](https://github.com/brick/math/releases/tag/0.3.3) - 2015-06-07

Fix toString() not handling negative numbers.

## [0.3.2](https://github.com/brick/math/releases/tag/0.3.2) - 2015-06-07

BigInteger and BigDecimal now have a getSign() method that returns:

* -1 if the number is negative
* 0 if the number is zero
* 1 if the number is positive

## [0.3.1](https://github.com/brick/math/releases/tag/0.3.1) - 2015-06-05

Minor performance improvements

## [0.3.0](https://github.com/brick/math/releases/tag/0.3.0) - 2015-06-04

The $roundingMode and $scale parameters have been swapped in BigDecimal::dividedBy().

## [0.2.2](https://github.com/brick/math/releases/tag/0.2.2) - 2015-06-04

Stronger immutability guarantee for BigInteger and BigDecimal.

So far, it would have been possible to break immutability of these classes by calling the unserialize() internal function. This release fixes that.

## [0.2.1](https://github.com/brick/math/releases/tag/0.2.1) - 2015-06-02

Added BigDecimal::divideAndRemainder()

## [0.2.0](https://github.com/brick/math/releases/tag/0.2.0) - 2015-05-22

* min() and max() do not accept an array anymore, but a variable number of parameters
* **minimum PHP version is now 5.6**
* continuous integration with PHP 7

## [0.1.1](https://github.com/brick/math/releases/tag/0.1.1) - 2014-09-01

* Added BigInteger::power()
* Added HHVM support

## [0.1.0](https://github.com/brick/math/releases/tag/0.1.0) - 2014-08-31

First beta release.