myCalculator,
Rewriting Java Math class

Guglielmo Del Sarto
in collaboration with Leonardo Scoppitto

October 2020

Indice

1 Basic Algebra

1.1 Extreme values

With the two functions:

```
public static double getMax( array ) {...}
public static double getMin( array ) {...}
```

they return the specified extreme value.

1.2 Algebraic sum

With the two functions:

```
public static double getSum( array ) {...}
public static double getSubtraction( array ) {...}
```

they return either sum of all elements of the array or subtraction of every elements with each oter.
When dealing with doubles, we implemented the Kahan compensated summation.

1.3 Product and division

With the two functions:

```
public static double getProduct( array ) {...}
```

you can have the product of an array values.

1.4 Percentage

With the function:

```
public static double getPercentage(value, percentage) {...}
```

you can get the percentage of a given double. You can either enter the decimal value (e. g. 0.3) or the integer (e. g. 30).

1.5 Absolute value

With the function:

```
public static double getAbs( x ) {...}
```

you can get the absolute value of x (it works with double and integers).

2 Powers and related

2.1 SquareRoot

With the function:

```
public static double getSqrt( x ) {...}
```

you can have the square root of x (double). It has been implemented with a really high precision (average Δ with respect to Math.sqrt() is on the order of 10^{-16}).

2.2 Power

With the function:

```
public static double getPower(base, exponent) {...}
```

you can have the $base^{exponent}$. The problem is that the function is not so accurate (see in the code) for very high numbers. However, relative error is still in the order of 10^{-15}.

2.3 Logarithm

With the two functions:

```
public static double getNatLog( x ) {...}
public static double getBaseLog(base, x) {...}
```

you can have either the natural logarithm of a given double or a logarithm given a certain base (average Δ with respect to Math.log() is on the order of 10^{-15}).

3 Sorting

3.1 QuickSort

With the two function:

```
public static void getArraySorted( array ) {...}
```

you sort the vector. It should be stressed that this change your original piece of memory and your input array is changed forever.
If you prefer to preserve your original array, you may use:

```
public static double[] getSortedArray( array ) {...}
```

which returns a sorted array without altering the original one!

4 Auxiliary functions

4.1 Arithmetic-Geometric mean

With the function:

```
public static double performAgm(x, y) {...}
```

you can have the AGM of two doubles.

4.2 Copy an array

With the function:

```
public static double[] performCopy( array ) {...}
```

you can copy an array in other piece of memory. It can be useful when you want to manipulate the array but do not want to alter the original array.