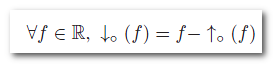
Notation:

The function that returns the rounded value of a real number:

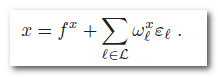


The function that returns the roundoff error is defined by:



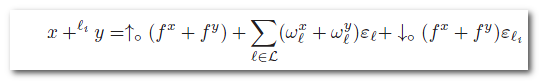
Expression:

Assume that the control points of a program are annotated by unique labels *l ∈ L*, and that *L* denotes the union of *L* and the special word *hi* used to denote all terms of order higher or equal to 2. A number *x* is represented by



Addition

The result of an arithmetic operation contains the combination of existing errors on the operands, plus a new roundoff error term. For addition and subtraction, the errors are added or subtracted componentwise :

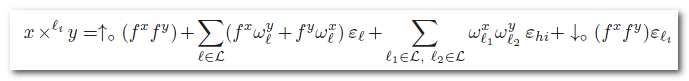


Subtraction

Same as addition

Multiplication

The multiplication introduces higher order errors, we write:

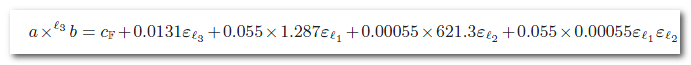


Example

Here is an introductory example in which we consider a simplified set F of floating-point numbers composed of a mantissa of four digits written in base 10.



We now consider the product *c* = *a × b,* The computation *a × b* in real numbers intended by the programmer is:



We keep only one term gathering the errors of order higher than one, and rewrite:

