Arithmetical Calculation with Roman Numerals and the Abacus[1]

John N. Martin
Department of Philosophy
University of Cincinnati
john.martin@uc.edu
April 17, 2006 (Corrections Jan. 17, 2011)

The purpose of this paper is to show how easy in fact it is to add, subtract, multiply and divide even large Roman numerals in ways that are isomorphic to the usual methods of carrying out these operations on the abacus. Roman numerals are regularly criticized as inappropriate for carrying out these calculating in an efficient way, and even historical claims are made about the administrative inefficiencies the numeral system imposed on Roman bureaucracy. The blame is placed on the lack of 0. These claims are all false. It is easy to represent all the positive rationals and to carry out the basic numerical operations quickly and efficiently, even without 0. Moreover, the system may easily be supplemented to include zero, the negative rationals and even decimal representations of the reals. First we must define clearly what a Roman numeral is. We shall do so hand in hand with definitions of the Arabic system so we can point out similarities and differences.

1. Definitions of the Numeral Systems

Basic Syntax. Both number systems work by representing multiple units of powers of 10. A single numeral, for example, may represent two 1's, three 10's, and seven 100's. In Arabic numerals this number is represented by 732. In Roman numerals it is named by DCCXXXII. In this section we will define the syntax and semantics of the numerals in their respective systems.

We begin by laying down some conventions about syntax, i.e. about how to talk about signs and strings of signs. Logicians (and linguists) call an individual symbolic letter or symbol a ***sign***. Included in signs are both individual letters of the alphabet and mathematical symbols. Here we shall be making use of a limited set of so-called "basic signs":

The ***basic signs for Arabic numerals*** (in their ***natural order*** <) are 0,1,2,3,4,5,6,7,8,9.
The ***basic sign for Roman numerals*** (in their ***natural order*** <) are I and V.

[1] For a discussion of the multiplication algorithm see Michael Detlefsen, Douglas K. Erlandson, J. Clark Heston, and Charles M. Young, "Computation with Roman Numerals, *Archive for History of Exact Sciences,* 15:2 (1976), 141-148. On the history of the numeral systems as described here see George Ifrah, *From One to Zero: A Universal History of Numbers* (New York: Viking Penguin, 1987 [1981]), Lowell Bair, trans.

We next introduce vocabulary for describing the result of combining individual signs to form longer expressions. Two signs s and s' are juxtaposed by first writing the sign s and then writing the sign s' to its immediate right. This process of writing one sign to the right of another is called ***concatenation.*** By repeating the process, a finite series of signs can be written one next to the other. Any finite concatenation of signs is called by linguists a ***string*.** Arabic and Roman numerals are strings composed of their respective basic signs.

To define these numerals precisely, we also need a way to label the position of a sign within a string and to count the number of places it is from the right hand terminus of the string. For this purpose we will use subscripts. By $e_n \ldots e_0$ we mean the result of concatenating the individual signs $e_n, \ldots, e_0$.
To define this idea more rigorously, let us introduce the symbol ^ to represent the operation of concatenation, i.e. $x\hat{}y = xy$. A special case is $x\hat{}x = xx$.

Let us also use the letter e with and without subscripts to range over signs. Using this notation, $e_i, \ldots, e_j$ are all signs and $\{ e_i, \ldots, e_j \}$ is a set of signs.

Then $e_n \ldots e_0$, the ***finite concatenation*** of $e_0, \ldots, e_0$, defined recursively as follows:

$e_1 e_0 = e_1\hat{}e_0$

$e_{n+1} \ldots e_0 = e_{n+1}\hat{}e_n \ldots e_0$

Each e_i in $e_n \ldots e_0$ is called an ***occurrence*** of e_i in $e_n \ldots e_0$, and it is possible that the same sign occur more than once. That is, it is possible that e_i be the same sign as e_j .

Note that in the notation $e_n \ldots e_0$ the role of subscripts in this notation is simply to indicate that e_i and e_j are different occurrences of a signs and to enumerate them, counting from the right. The subscript is not part of the sign itself.

Examples

$a\hat{}b\hat{}c = abc$
$a\hat{}b\hat{}a=aba$
$a\hat{}a\hat{}a=aaa$
a_0=a
$a_2b_1c_0= a\hat{}b\hat{}c = abc$
$a_2b_1a_0= a\hat{}b\hat{}a=aba$
$a_2a_1a_0= a\hat{}a\hat{}a=aaa$

The individual signs that make up numerals in both number systems are marked by features that indicate what power of 10 they represent. In Arabic numerals this feature is the number of the position they hold counting from the right. In Roman numerals position is not important. Rather the basic signs themselves change depending on what power of ten they represent. One way this is done is by introducing an entirely new symbol for each order of 10 to stand for a single unit of that order, e.g. I stands for a single unit of 1's (for 10^0), to X for a unit of 10's (for 10^1), to C for a unit of 100's (for 10^2), to M for a unit of 1000's (for 10^3).   However, since there are an infinite number of orders of 10, and we have only finite capacities, we cannot continue to invent a new symbol for each order. Some other devise must be introduced to represent higher powers. Over the centuries the users of Roman numerals invented several notations for this purpose.

Perhaps the best known is the vinculum (or *titulus* in mediaeval Latin) that consists of a superscripted bar over a numeral *n.* The bared *n* stands for 1000 *n*'s.

$\overline{\mathrm{D}}$=500,000
$\overline{\mathrm{M}}$= 1,000,000
$\overline{\mathrm{X}}\overline{\mathrm{V}}\overline{\mathrm{I}}$= 18,000

If the vinculum turned down at the edges the subtended number *n*, the whole represents 100,000 units of *n*:

$\overline{|\mathrm{XV}|}$=1,500,000

$\overline{|\mathrm{XV}|}\,\overline{\mathrm{X}}\overline{\mathrm{V}}\overline{\mathrm{I}}$CCLII =1,518,252

In practice the Romans did not use multiple bar superscripts above a single letter.

A similar device of flanking curves or "legs" was adapted from Etruscan numeration. Each pair of symmetric legs indicates a multiple of 1,000.   It is this notation that gave rise to the use of M for 1000 (it is not from *mille*) and D for 500:

$(\mid) = (\mid) = \Phi$ = M = 1,000
$\mid) = \ \mid)$ = D = 500
$((\mid)) = ((\mid))$ = 10,000
$(((\mid))) = (((\mid)))$ = 100,000

Both notational conventions are found in mediaeval usage.

A more general convention was a special device of prefixing a lower number *n* to a higher number *m* to obtain the concatenated form *nm.* (Note that this notation is not to be confused with so-called subtractive notation of IV for IIII, IX for VIIII, XC for LXXXX etc., which was not standardized until relatively recent centuries.) Here *nm* stands for the Roman numeral that consists of writing *m n* times, i.e. it stands for $n \times m$. Some authors use the convention that the second and higher number is written as a super or subscript:

XVMDXXI = XV$^{\mathrm{M}}$DXXI = 15,521

This notation is in fact equivalent to the superscript notation introduced below since any I^n or V^n may be written as I*n** or *Vn** (or as *n**V if V is greater than *n**) such that *n** is 10 raised to the *n*th power. With the addition of this notation, it follows that every positive integer has at least one finitely long Roman numeral that names it, and we shall state this result more formally shortly.

Since our purpose in these notes is to explore how easy it is, in general, to add, subtract, multiply and divide Roman numerals, in a way similar to the normal methods for doing these operations on the abacus, it will be convenient to use a simpler notation to represent units and fives of the various powers of 10. Rather than superscripts let us modify the vinculum notation. If there is no bar then let us understand a numeral as standing for units of the 0 power of 10. If there are *n* bars, let it stands for units of the *n*th power of 10. For example:

$I^1 = \overline{I} = X$
$V^1 = \overline{V} = L$
$I^2 = \overline{\overline{I}} = \overline{X} = C$
$V^2 = \overline{\overline{V}} = \overline{L} = D$
$I^3 = \overline{\overline{\overline{I}}} = \overline{\overline{X}} = \overline{C} = M$
$V^3 = \overline{\overline{\overline{V}}} = \overline{\overline{L}} = \overline{D}$

Moreover, let us define the notation in a way general enough to apply to Arabic numerals as well:

An ***indexed basic sign*** is defined as any $\overset{n}{e}$ such that *e* is a basic numeral (Arabic or Roman) and *n* is a series (possibly empty) of horizontal bars.[2]

In the case of both Arabic and frequently even Roman numerals, we will rewrite $\overset{n}{e}$ as e^n and replace the *n* with the Arabic numeral for the number of bars in *n.* Thus, $\overline{\overline{1}}$ will be rewritten 1^2 and frequently $\overline{\overline{I}}$ will be rewritten I^2.

According to this convention, if there are no bar at all over *e*, then *e* represents units in the 0 power of 10, i.e. units; one bar means that *e* represents units in the 1st power of 10, i.e. tens; two bars indicates it represents the 2nd power of 10, 100's (i.e. 10^2), etc. In principle there could be any finite number of bars above a letter, If there were 2534 bars above I, then the numeral would represent of units of 1^{2534}. Mathematically, the use of bars has the nice feature that it allows us to avoid the circle of having to appeals to numerals as superscripts in the very *definiens* of *numeral.* It should be stressed that this notation departs from the historical titulus notation, which represented multiples of 1000 and which did not employ multiple bars over a single numeral.

Let us now complete the syntax for the core notion of numeral by presenting the actual definition of a numeral in its respective system. We do so by combining in a single notation a finite concatenation of indexed basic signs.

[2] Here is a more formal recursive definition of $\overset{n}{e}$:

$\overset{1}{e} = \overline{e}$
$\overset{n+1}{e} = \overline{\overset{n}{e}}$

An ***indexed Arabic/Roman numeral (in non-ordered form)*** is defined as any *n* place concatenation $e^i{}_n \ldots e^k{}_0$ of indexed basic Arabic/Roman signs $e^k, \ldots, e^i$.

We extend the notion of ***natural order*** ≤ to indexed numerals by ranking one of two numerals above the other if its index is higher, or if their indexes are th same, and ranks prior to having an index, or if both numerals have the same index and neither ranks higher than the other prior to having an index (which happens only if the two basic numerals are same).

For any indexed numeral, $x^i \leq y^j$ iff, either

1. $i<j$,
2. $i=j$ and $y<x$, or
3. $i=j$ and $x=y$.

Examples

$\overline{\mathrm{X}} \leq \overline{\overline{\mathrm{X}}}$ because 1<2
$\overline{\mathrm{I}} \leq \overline{\mathrm{V}}$ because I<V
$\overline{\mathrm{I}} = \overline{\mathrm{I}}$ because I = I
$4 \leq \overline{4}$ because 0<1
$\overline{2} \leq \overline{4}$ because 2<4
$\overline{2} = \overline{2}$ because 2=2

Basic Semantics. To “interpret” these number systems, we need to assume that we can talk about the natural numbers. In doing so we must be careful that we do not assume that we already understand the numeral systems we are defining. In particular we must not assume we already understand the Arabic numeral system. We can avoid this circle by referring to the natural numbers as they are constructed in set theory. It is possible to define the set *Nn* of natural numbers, and a series **0**,**1**,**2**,**3**,**4**,… of names for them, without presupposing the definition of the use of the Arabic system. The details are technical and may simply be assumed here.[3] We will also assume that the identity relation **=** and the operations of ***s*** (successor), **+** (addition) and • (multiplication) on *Nn* are well defined. Note that these too are definable in set theory. Here ***s***(*n*) stands for the successor *n***+**1 of the natural number *n*. For convenience and to aid understanding, we shall use below the bold face notation **0**,**1**,**2**,**3**,**4**,… as abbreviations of the set theoretic names of the natural numbers, e.g. **0** is short for zero defined in set theory, namely it is short for ∅ (the empty set), and **1** fis short for {∅ }. Note that these bold face symbols are *not* the Arabic numerals, but

[3] First define **0** as ∅. Next define the successor operation ***s*** as follows: for any *x* , ***s***(*x*)=*x*∪{*x*}. Then the set of natural numbers *Nn* is defined as the least set *A* such that **0** is in *A* and for any *x*, if *x* is in *A*, then so is ***s***(*x*). We then define individual names for members of *Nn:* **1** is ***s***(**0**), **2** is ***s***(**1**), **3** is ***s***(**2**), etc.

abbreviations for set theoretic terms defined independently of the Arabic numeral system.

With these assumptions we now define $\Im$, ***the standard interpretation***, for each system.

Recursive Definition of $\Im$, *the standard interpretation*, for indexed Arabic numerals

$\Im(0^0) = \mathbf{0}$
$\Im(1^0) = \mathbf{1}$
$\Im(2^0) = \mathbf{2}$
$\Im(3^0) = \mathbf{3}$
$\Im(4^0) = \mathbf{4}$
$\Im(5^0) = \mathbf{5}$
$\Im(6^0) = \mathbf{6}$
$\Im(7^0) = \mathbf{7}$
$\Im(8^0) = \mathbf{8}$
$\Im(9^0) = \mathbf{9}$
$\Im(e^{n+1}) = \Im(e^n)\bullet\mathbf{10}$

Recursive Definition of $\Im$, *the standard interpretation*, for indexed Roman numerals

$\Im(\mathrm{I}^0) = \mathbf{1}$
$\Im(\mathrm{V}^0) = \mathbf{5}$
$\Im(e^{n+1}) = \Im(e^n)\bullet\mathbf{10}$

Theorem. $\Im(e^{n+1}) = \Im(e^n)\bullet\mathbf{10} = \mathbf{1}^n\bullet\Im(e)$

Examples

$\Im(4^3) = \Im(4^2)\bullet\mathbf{10}=\Im(4^1)\bullet\mathbf{10}\bullet\mathbf{10} = \Im(4^0)\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10} = \mathbf{4}\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10} = \mathbf{4000}$
$\Im(\mathrm{V}^3) = \Im(\mathrm{V}^2)\bullet\mathbf{10}=\Im(\mathrm{V}^1)\bullet\mathbf{10}\bullet\mathbf{10} = \Im(\mathrm{V}^0)\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10} = \mathbf{5}\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10} = \mathbf{5000}$

Definition of $\Im$ for complex Arabic/Roman numerals:

$\Im(e^i{}_n\ldots e^k{}_0) = \Im(e^i{}_n)+\ldots+\Im(e^k{}_0)$

Examples

$\Im(1^2) = \mathbf{1}\bullet\mathbf{10}\bullet\mathbf{10} = \mathbf{100}$
$\Im(2^3) = \mathbf{2}\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10} = \mathbf{2000}$
$\Im(2^3{}_1 1^2{}_0) = \Im(2^3)+\Im(1^2) = (\mathbf{2}\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10}) + (\mathbf{1}\bullet\mathbf{10}\bullet\mathbf{10}) = \mathbf{2100}$
$\Im(\mathrm{I}^2) = \mathbf{1}\bullet\mathbf{10}\bullet\mathbf{10} = \mathbf{100}$
$\Im(\mathrm{V}^3) = \mathbf{1}\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10}) = \mathbf{5000}$
$\Im(\mathrm{I}^2{}_1\mathrm{V}^3{}_0) = \Im(\mathrm{I}^2)+\Im(\mathrm{V}^3) = (\mathbf{1}\bullet\mathbf{10}\bullet\mathbf{10})+(\mathbf{1}\bullet\mathbf{10}\bullet\mathbf{10}\bullet\mathbf{10}) = \mathbf{1500}$

Theorem. For any non-zero natural number ***n*** there exists a roman numeral *n* such that $\Im(n) = \boldsymbol{n}$.

For convenience we will introduce the identity sign = so that we can write "equation" using Arabic and Roman numerals. The equations $n=m$ is true in the standard interpretation $\Im$ if and only if $\Im$ assigns to the two numerals n and m the identical natural number:

$\Im(n=m) = \mathrm{T}$ iff $\Im(n)=\Im(m)$

Rationals and Real Numbers. It should be remarked that it is no more difficult in Roman than Arabic numerals to extend the system to represent positive rational numbers, zero, the negative rationals and the reals. This is true since every rational may be named by decimal notation, which may be used in the Romans as well as the Arabic system. In the Roman numeral system all we need do is to introduce notation for representing units of the negative powers of 10. The notation readily lends itself to doing so though the Romans and mediaeval users did use the Roman numeral system to name the fractions between 1 and 0 but rather used specific names for commonly used factions, e.g. *one half* for ½,, *a third* for .333…. and *a quarter* for ¼.[4] These extensions to not provide a symbol for 0. However, as will be clear below when we discuss the abacus, 0 may be regarded as a completely blank abacus (with no active beads) or by what linguists call the empty string, i.e. a string composed of zero signs. As will be clear, it remains possible on this understanding of how to represent zero to perform the standard operations of addition, subtraction, multiplication and division.

Complex Numerals. Complex numerals as defined so far need not be arranged with their component basic sings ordered in increasing value from right to left. Reordering them in this way has two advantages. First it makes them easier to read. In the case of Arabic numerals it also has the advantage that if we introduce 0 as a place holder for powers of 10 that are not represented then the actual distance of a basic sign from the right correlates to the power it represents, thus permitting us to do away with its superscript index.

To effect this simplification let us first reorder numerals in increasing value (in their "natural order" $\leq$) from right to left.

Definition of Ordered Form

---

[4] Consider just positive decimals. The background theory would need to be extended to include the set Z^+ of positive rationals, the relation $\leq$, and the operations + and $\bullet$, as usually defined. An ***indexed basic sign*** would need to be redefined as any $\overset{n}{e}$ or $\underset{n}{e}$ such that e is a basic numeral (Arabic or Roman) and n is a series (possibly empty) of horizontal bars. Again $\overset{n}{e}$ is written e^n and $\underset{n}{e}$ is e_n. The definition of $\Im$ is revised to map indexed basic signs into Z^+ with the new clause: $\Im(e_{n\text{-}1}) = \Im(e_n)\bullet\mathbf{1/10}$ It will then follow that for any positive rational n there exists a roman numeral n such that $\Im(n) = \boldsymbol{n}$. The 3.1417 would be written $\mathrm{I}^0\mathrm{I}^0\mathrm{I}^0\mathrm{I}^{-1}\mathrm{I}^{-2}\mathrm{I}^{-2}\mathrm{I}^{-2}\mathrm{I}^{-2}\mathrm{I}^{-3}\mathrm{V}^{-4}\mathrm{I}^{-4}\mathrm{I}^{-4}$, which could be reformulated in bar notation.

$e^i{}_n \ldots e^k{}_0$ is in ***ordered form*** if and only if for $e^x{}_m$ and $e^y{}_l$ in $\{e^i{}_n,\ldots,e^k{}_0\}$ if $m \leq l$, then $e^y{}_l \leq e^x{}_m$

Theorem. For any finite set $\{e^i{}_n,\ldots,e^k{}_0\}$ of indexed Arabic/Roman numerals there is one and only one complex numeralArabic/Roman, call it $\boldsymbol{or}[e^i{}_n \ldots e^k{}_0]$, in ordered form.

Examples

$\boldsymbol{or}[3^1{}_3 6^3{}_2 7^5{}_1] = 7^5{}_3 6^3{}_2 3^1{}_1$
$\boldsymbol{or}[3^2{}_3 6^3{}_2 4^2{}_1] = 6^3{}_3 4^2{}_2 3^2{}_1$
$\boldsymbol{or}[3^2{}_3 6^3{}_2 3^2{}_1] = 6^3{}_3 3^2{}_2 3^2{}_1$

$\boldsymbol{or}[\mathrm{I}^1{}_3 \mathrm{V}^3{}_2 \mathrm{I}^5{}_1] = \mathrm{I}^5{}_3 \mathrm{V}^3{}_2 \mathrm{I}^1{}_1$
$\boldsymbol{or}[\mathrm{I}^2{}_3 \mathrm{V}^3{}_2 \mathrm{V}^2{}_1] = \mathrm{V}^3{}_3 \mathrm{V}^2{}_2 \mathrm{I}^2{}_1$
$\boldsymbol{or}[\mathrm{I}^2{}_3 \mathrm{V}^3{}_2 \mathrm{I}^2{}_1] = \mathrm{V}^3{}_3 \mathrm{I}^2{}_2 \mathrm{I}^2{}_1$

Though the indices in an ordered form are ranked in increasing value from right to left, as so far defined a string may contain more than one basic numeral of the same index value. To find a simpler equivalent, let us eliminate ("collapse") such a numeral into a shorter numeral that represents the same natural number. In an Arabic numeral within a given power of 10, let us replace, 1+1 by 2, 2+1 by 3, etc. up to 8+1 by 9. Let us also replace 9+1 of one power by 1 of the next power. In a Roman numeral let us replace within a given power of 10 a series of five I's by a V of that power, and two V's of a power by one I of the next higher power. Given the standard interpretation, it is a simple truth of arithmetic that the result will name an identical number:

Theorem

$e^i{}_n \ldots \mathrm{I}^{n+1}{}_i \ldots e^k{}_0 = e^i{}_{n+1} \ldots \mathrm{V}^n{}_{i+1} \mathrm{V}^n{}_i \ldots e^k{}_0$
$e^i{}_n \ldots \mathrm{V}^n{}_i \ldots e^k{}_0 = e^i{}_{n+4} \ldots \mathrm{I}^n{}_{i+4} \mathrm{I}^n{}_{i+3} \mathrm{I}^n{}_{i+2} \mathrm{I}^n{}_{i+1} \mathrm{I}^n{}_i \ldots e^k{}_0$

Definition of (Non-Redundant) Collapsed Roman Numerals

For any string $s = e^i{}_m \ldots e^k{}_0$, $\boldsymbol{clps}[e^i{}_m \ldots e^k{}_0]$, the **collapsed form** of ***s*** is the unique string s' in ordered form (i.e. $s' = \boldsymbol{or}[s]$) such that for no n,u,v,l is it the case that either $s \fallingdotseq e^u{}_{l+1} \ldots \mathrm{V}^n{}_{i+1} \mathrm{V}^n{}_i \ldots e^v{}_0$, or $s \fallingdotseq e^u{}_{l+4} \ldots \mathrm{I}^n{}_{i+4} \mathrm{I}^n{}_{i+3} \mathrm{I}^n{}_{i+2} \mathrm{I}^n{}_{i+1} \mathrm{I}^n{}_i \ldots e^v{}_0$.

Theorem. For any Arabic/Roman numeral $e^i{}_n \ldots e^k{}_0$ there is one and only one $\boldsymbol{clps}[e^i{}_m \ldots e^k{}_0]$ and it is an Arabic/Roman numeral in ordered form.

(Proof is by induction.)

Theorem. For any Arabic numeral $e^i{}_n \ldots e^k{}_0$,

$$\Im(e^i{}_n \ldots e^k{}_0) = \Im(\boldsymbol{or}[e^i{}_n \ldots e^k{}_0]) = \Im(\boldsymbol{clps}[e^i{}_n \ldots e^k{}_0])$$

In Roman numerals it often happens that there are multiple basic numerals with of the same index. This happens in Roman numerals like $\mathrm{V}^0{}_2 \mathrm{I}^0{}_1 \mathrm{I}^0{}_0$, which contains $I^0{}_1 I^0{}_0$. It is for that reason that each basic expression in Roman numerals must retain its index information, either as bar notation or in some other way. It is customary, however, to simplify this information by representing it in a series of abbreviated forms. The familiar symbols below abbreviate the basic numerals of the early indices of I and V:

Definitions

$e^i{}_n \ldots X_i \ldots e^k{}_0 = e^i{}_n \ldots I^1{}_i \ldots e^k{}_0$
$e^i{}_n \ldots L_i \ldots e^k{}_0 = e^i{}_n \ldots V^1{}_i \ldots e^k{}_0$
$e^i{}_n \ldots C_i \ldots e^k{}_0 = e^i{}_n \ldots I^2{}_i \ldots e^k{}_0$
$e^i{}_n \ldots D_i \ldots e^k{}_0 = e^i{}_n \ldots V^2{}_i \ldots e^k{}_0$
$e^i{}_n \ldots M_i \ldots e^k{}_0 = e^i{}_n \ldots I^3{}_i \ldots e^k{}_0$

Since, however, X,L,C,D,M only go so far, the bar notation is extended to them. This is possible because they abbreviate bared basic numerals for which the bar notations defined. Thus, the identities below follow from the definitions above and bar notation.

Theorem

$\mathrm{I}^1 = \overline{\mathrm{I}} = \mathrm{X}$
$\mathrm{V}^1 = \overline{\mathrm{V}} = \mathrm{L}$
$\mathrm{I}^2 = \overline{\mathrm{X}} = \mathrm{C}$
$\mathrm{V}^2 = \overline{\overline{\mathrm{V}}} = \overline{\mathrm{L}} = \mathrm{D}$
$\mathrm{I}^3 = \overline{\overline{\overline{\mathrm{I}}}} = \overline{\overline{\mathrm{X}}} = \overline{\mathrm{C}} = \mathrm{M}$
$\mathrm{V}^3 = \overline{\overline{\overline{\mathrm{V}}}} = \overline{\overline{\mathrm{L}}} = \overline{\mathrm{D}}$

Since each component basic sign of a complex Roman numeral retains its index, once the components have be put in ordered form and collapsed, the subscript notation conveys no information, and may be simply deleted.

Theorem

For any Roman numeral, $e^i{}_n \ldots e^k{}_0$ if ***clps***$[e^i{}_n \ldots e^k{}_0] = e^u{}_m \ldots e^v{}_0$ then $e^u{}_m \ldots e^v{}_0$ is the finite contamination $e^u \ldots e^v$ and ***clps***$[e^i{}_n \ldots e^k{}_0] = e^u \ldots e^v$.

Given that it is not possible in general to represent the index of a component of a Roman numeral by its position in the right to left rank, the custom arose to abbreviate longer strings of components within a Roman numeral in a simpler so-called "subtractive" notation, for example, IV for IIII, XL for XXXX, DM for DCCCCL, etc.

Definition.

$I^n I^{n+1} = V^n I^n I^n I^n I^n$
$I^n V^{n+1}{}_i = I^n I^n I^n I^n$
$V^n I^{n+1} = V^n I^n I^n I^n I^n$

Example

$$\begin{aligned}\Im(V^2I^3I^0V^1) &= \Im(I^3V^2V^1I^0)\\ &= \Im(I^3I^2I^2I^2I^2I^2I^1I^1I^1I^1I^1I^0)\\ &= \Im(I^3)+\Im(I^2)+\Im(I^2)+\Im(I^2)+\Im(I^2)+\Im(I^2)+\Im(I^1)+\Im(I^1)+\\ &\quad \Im(I^1)+\Im(I^1)+\Im(I^1)+\Im(I^0)\\ &= \mathbf{1551}\end{aligned}$$

However, because subtractive abbreviations introduce a second value for I and V, making them ambiguous as a function of their position, it prevents the automatic application of the normal Roman numeral algorithms for addition and multiplication. It is best, therefore, to do without it in computation. In practice doing so means eliminating all subtractive abbreviation before doing any computation, and then if necessary, reintroducing them afterwards.

Example. Let e^i be I^3 and e^k be I^2. Then, $e^i{}_1e^k{}_0$ is e^ie^k, which is $I^3\,I^2$.

Abbreviations within Arabic Numerals

Let us now return to Arabic numberals and explain how they are to be collapsed. By making use of the basic fact arithemetic that a number from 1 to 9 is identical to the successor of its predecessor, any the constant numeral for a number may replace any series of earlier numbers in the series that sum to it.

Theorem

$$\begin{aligned}
e^i{}_n\ldots 2^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 1^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots 3^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 2^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots 4^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 3^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots 5^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 4^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots 6^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 5^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots 7^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 6^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots 8^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 7^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots 9^n{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 8^n{}_{i+1}1^n{}_i \ldots e^k{}_0\\
e^i{}_n\ldots I^{n+1}{}_i \ldots e^k{}_0 &= e^i{}_{n+1}\ldots 9^n{}_{i+1}1^n{}_i \ldots e^k{}_0
\end{aligned}$$

Definition of (Non-Redundant) Collapsed Arabic Numerals

clps$[e^i{}_n\ldots e^k{}_0]$, the **collapsed form** of a string $e^i{}_n\ldots e^k{}_0$, is the unique string s in ordered form, if it exists, $\Im(s)=\Im($***or***$[e^i{}_n\ldots e^k{}_0]\,)$ and is such that for all $e^x{}_i$ and $e^y{}_j$ for $i,j=1,\ldots,n$, if $i\neq j$, the $x\neq y$.

Theorem. For any Arabic numeral $e^i{}_m\ldots e^k{}_0$, ***clps***$[e^i{}_m\ldots e^k{}_0]$ exists, is an Arabic numeral, and is unique.

(Proof is by induction.)

Theorem. For any Arabic numeral $e^i{}_m\ldots e^k{}_0$, $\Im(e^i{}_n\ldots e^k{}_0)$ **=** $\Im($***or***$[e^i{}_n\ldots e^k{}_0]\,)$ **=** $\Im($***clps***$[e^i{}_n\ldots e^k{}_0])$

The Placeholder 0 in Arabic Numerals

Thus far, the value of a basic numeral within a complex numeral is indicated only by its superscripted index. In general, a signs index is not the same as its rank in the left to right order, even if the numeral is in ordered form. This happens because, even if the signs in a numeral are ordered, there may be some power of 10, i.e. some index value, that is not represented. For example in $3^2{}_1 4^0{}_0$, there is no sign with index 1, which would represent 10's. As a consequence, the superscript and the subscript of the components are not the same. Now, if every power of 10 is represented in a numeral, two things follow:

(1) $e^i{}_n \ldots e^k{}_0$ is in ordered form and

(2) in every case the superscript and the subscript are the same.

If both these condition hold, then we can drop the superscripts and subscripts, because we can tell from a sign's rank what power of 10 it represents. We count from the right. A sign on rank *n* represents units of the *n*-the power of 10. To do this however in for every number there must be some sign occupying every rank, even for those powers of 10 that have no value listed. It was for this purpose that the Arabs invented 0. In order to permit the dropping of super and subscripts, let us introduce placeholder $0^n{}_m$ for missing powers *n.* That is, $e^i{}_n \ldots e^k{}_0$ lacks a component $e^n{}_m$, we insert $0^n{}_m$ and reorder. This placeholder $0^n{}_m$ represents zero units of the power of 10. That is, it stands for **0.**

To insert in the string $e^i{}_n \ldots e^j{}_0$ we first collapse $e^i{}_n \ldots e^j{}_0$ to $e^w{}_l \ldots e^z{}_0$. We now identify these powers *u,…,v* not represented in $e^w{}_l \ldots e^z{}_0$. Supose there are *m*+1 of these. We then constructe *m+1* sings represent zero value $e^i{}_n \ldots e^j{}_0$, concatenate them to $e^w{}_l \ldots e^z{}_0$, and order the result:

pl[$e^i{}_n \ldots e^j{}_0$], ***the placeholder form*** of an Arabic numeral $e^i{}_n \ldots e^j{}_0$ is defined as ***or***[$e^w{}_{l+m+1} \ldots e^z{}_{0+m+1} e^u{}_m \ldots e^v{}_0$] such that

1. ***clps***[$e^i{}_n \ldots e^j{}_0$] = $e^w{}_l \ldots e^z{}_0$, and
2. {$e^u{}_m \ldots e^v{}_0$} is { $0^x{}_k$ | $k \le i$, and there is no $e^x{}_y$ such that $e^x{}_y$ in {$e^i{}_n \ldots e^j{}_0$}}.

Since ***pl*** simply inserts 0's into a string series with for positions with superscripts and then renumbers, it is well-defined for any series and yields a unique output. Conversely, deleting the 0's and renumbering restores the original series. This is true even if the series has not be reordered or collapsed. It follows that ***pl*** is well defined and biunique for all Arabic numerals.

Theorem

For any Arabic numeral $e^i{}_n \ldots e^k{}_0$ such that ***pl***[$e^i{}_n \ldots e^k{}_0$] = $e^u{}_m \ldots e^v{}_0$, $e^u{}_m \ldots e^v{}_0$ is the finite concatenation $e^u \ldots e^v$ and ***pl***[$e^i{}_n \ldots e^k{}_0$] = $e^u \ldots e^v$.

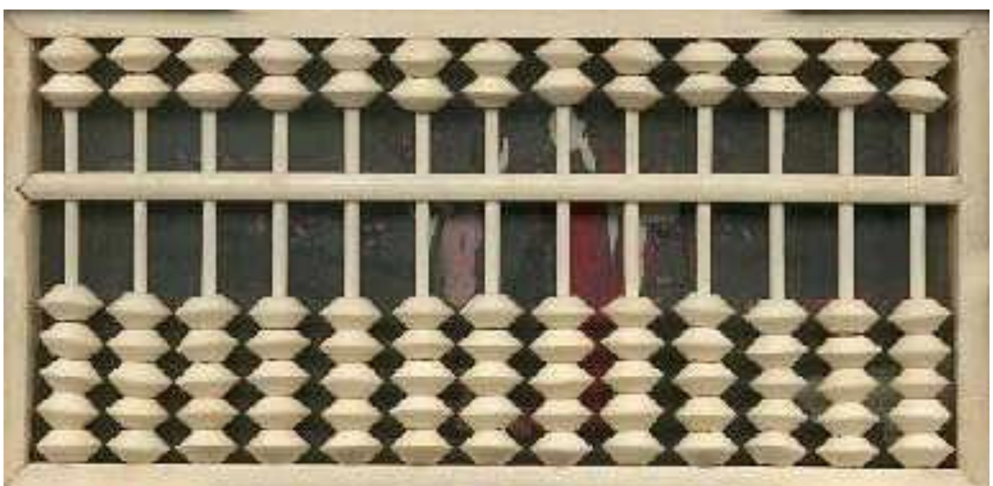
Example. An Arabic Numeral with 0. Consider $4^2{}_4\, 2^4{}_3\, 3^0{}_2\, 5^0{}_1 7^1{}_0$. Reorded, it becomes $2^4{}_4 4^2{}_3 7^1{}_2 3^0{}_1 5^0{}_0$. This collapses to $2^4{}_3 4^2{}_2 7^1{}_1 8^0{}_0$. This number lacks a component representing value of the 3rd power of 10, and thus requires a placeholder for that power. The placeholder form is then $2^4{}_4 0^3{}_3 4^2{}_2 7^1{}_1 8^0{}_0$. If we simplify deleting super and subscripts, this becomes 20478.

Example

$$
\begin{aligned}
\Im(4^2 2^3 3^0 7^1) &= \Im(2^3 4^2 7^1 3^0) \\
&= \Im(2^3)+\Im(4^2)+\Im(7^1)+\Im(3^0) \\
&= \mathbf{20000} + \mathbf{0} + \mathbf{400} + \mathbf{70} + \mathbf{3} \\
&= \mathbf{20473}
\end{aligned}
$$

2. The Abacus

It is helpful to think of indexed notation e^n in terms of an abacus. In the notation e^n, the superscript n indicates a row in an abacus: 0 names the rightmost row (units), 1 the row to its left (hundreds), 2 the next row left (thousands), etc. The numeral e indicates how many beads on that row you move. An abacus for Arabic numerals of the sort we happily never see beyond the second grade is moronically simple. It has 10 beads on parallel rows (plus perhaps an eleventh bead for "carrying"). A bead is in resting position if it is pushed down, and in active position if pushed up. Each row from right to left represents increasing powers of ten, and each bead on a row in active position represents one unit of that power.



A Chinese Abacus[5]

An abacus for Roman numerals (which is the usual sort of abacus, including those used in Japan and China) there are likewise a series of parallel rows from right to left representing the increasing powers of ten. However each row is divided into to parts, a lower part that contains four beads (and on many Chinese abaci there is also a fifth bead for "carrying"), and an upper part containing a single bead (and often a second bead for "carrying"). Lower beads are active when pushed up, and upper beads when pushed down. Each beads in the lower part represents one unit of the power of 10 represented by its row, and each upper bead represents five units of that power.

---

[5] Image from:
http://www.hh.schule.de/metalltechnik-didaktik/users/luetjens/abakus/china/china.htm

To represent e^n on an Arabic abacus, you simply push e beads into the active position (up) on row n+1. For example, the Arabic numeral 2473 in indexed notation is $2^3{}_3 4^2{}_2 7^1{}_1 3^0{}_0$, or

$1^3{}_{17}1^3{}_{16}1^3{}_{15}1^2{}_{14}1^2{}_{13}1^2{}_{12}1^2{}_{11}1^1{}_{10}1^1{}_8 1^1{}_7 1^1{}_6 1^1{}_5 1^1{}_4 1^1{}_3 1^0{}_2 1^0{}_1 1^0{}_0$,
$1^3 1^3 1^3 1^2 1^2 1^2 1^2 1^1 1^1 1^1 1^1 1^1 1^1 1^1 1^0 1^0 1^0$

Each 1^n represents one bead in the active position on the n^{th} row. Start at the right and move down the basic numerals from 0 to 17, moving one bead up on the row indicated by the superscript. Conversely, if you find the beads on the abacus arrayed in the form

$1^3 1^3 1^3 1^2 1^2 1^2 1^2 1^1 1^1 1^1 1^1 1^1 1^1 1^1 1^0 1^0 1^0$, or
$1^3{}_{17}1^3{}_{16}1^3{}_{15}1^2{}_{14}1^2{}_{13}1^2{}_{12}1^2{}_{11}1^1{}_{10}1^1{}_8 1^1{}_7 1^1{}_6 1^1{}_5 1^1{}_4 1^1{}_3 1^0{}_2 1^0{}_1 1^0{}_0$,

you can determine ("read") that it means what we call in Arabic notation 2473.

Now lets represent e^n on a Roman (or Japanese or Chinese) abacus. If e^n is, for example, the Roman numeral CCV or $C_2C_1V_0$ abbreviates $I^2{}_2 I^2{}_1 V^1{}_0$ or $I^2 I^2 V^1$. Each I^n represents one bottom bead on row n in active position (up), and each V^n represents one top bead on row n in active position (down). Thus $I^2 I^2 V^1$ is represented by moving two lower beads up on row 2 (third from the right) and one top bead down on row 1 (second from right).

3. Addition in Arabic and Roman Numerals, Roman Style

The interpretation of the 2-place ***addition operation*** + is defined as follows:

For any Arabic/Roman numeral,
$\Im(a^i{}_n \ldots a^k{}_0 + b^i{}_m \ldots b^k{}_0) \mathbf{=} \Im(a^i{}_n \ldots a^k{}_0) \mathbf{+} \Im(b^i{}_m \ldots b^k{}_0)$

From this definition a series of useful elementary theorems follow by elementary arithmetic:

Theorem

1. If $a^i{}_n \ldots a^k{}_0$ and $b^i{}_m \ldots b^k{}_0$ are Roman numerals, then
   $a^i{}_n \ldots a^k{}_0 + b^i{}_m \ldots b^k{}_0 = \textbf{\textit{clps}}[a^i{}_{n+m+1} \ldots a^k{}_{m+1} b^i{}_m \ldots b^k{}_0)]$
2. If $a^i{}_n \ldots a^k{}_0$ and $b^i{}_m \ldots b^k{}_0$ are Arabic numerals, then
   $a^i{}_n \ldots a^k{}_0 + b^i{}_m \ldots b^k{}_0 = \textbf{\textit{pl}}[a^i{}_{n+m+1} \ldots a^k{}_{m+1} b^i{}_m \ldots b^k{}_0)]$

Examples

51+25	=	(50 + 1) + (20 + 5)
	=	50+20+5+1
	=	70+6
	=	76

47 + 9 + 8	=	40 + 7 + 9 + 8
	=	40+24

	=	60 + 4
	=	64

Addition Table for Roman Numerals:

+	I^m	V^m
I^n	$I^n I^m$	$V^m I^n$
V^n	$V^n I^m$	$V^n V^m$

Examples

LI + XXV	=	$V^1 I^0 \; I^1 I^1 V^0$
	=	$V^1 I^1 I^1 V^0 I^0$
	=	LXXVI
LI + XXV	=	LIXXV
	=	LXXVI
XLVII + IX + VIII	=	XXXXVII VIIII VIII
	=	XXXX ~~VV~~ V IIIII IIIII
	=	~~XXXXX~~ V ~~IIIII~~ IIII
	=	L ~~VV~~ IIII
	=	L X IIII
XLVII + IX + VIII	=	XXXXVII VIIII VIII
	=	$I^1 I^1 I^1 I^1 V^0 I^0 I^0 \; V^0 I^0 I^0 I^0 I^0 \; V^0 I^0 I^0$
	=	$I^1 I^1 I^1 I^1 \; \cancel{V^0}\cancel{V^0} V^0 \; I^0 I^0 I^0 I^0 I^0 \; I^0 I^0 I^0 I^0$
	=	$\cancel{I^1}\cancel{I^1}\cancel{I^1}\cancel{I^1}\cancel{I^1} V^0 \; \cancel{I^0}\cancel{I^0}\cancel{I^0}\cancel{I^0}\cancel{I^0} \; I^0 I^0 I^0 I^0$
	=	$V^1 \cancel{V^0}\cancel{V^0} \; I^0 I^0 I^0 I^0$
	=	$V^1 \; I^1 \; I^0 I^0 I^0 I^0$
	=	L X IIII

Addition and Subtraction on the Abacus

No doubt the reason the abacus exists and has been used apparently independently by such diverse cultures as the ancient Romans, mediaeval Europe, and both modern and ancient oriental civilizations is the fact that it makes addition both simple and mindless.[6] To $a^i_n \ldots a^k_0 + b^i_m \ldots b^k_0$ simple enter $a^i_n \ldots a^k_0$ on the abacus, then enter on top of it $b^i_m \ldots b^k_0$ by first entering b^k_0, then b^k_1, then b^k_2, in order until you finally enter b^i_m. The only difficulty is that you may be required to move too many beads in a row into the active position than the row possess. In that case you "collapse" or "carry."

A Roman Abacus[7]

On a Roman abacus, you convert 5 active beads in the lower portion of row *n,* into 1 active bead in the upper portion of *n*, and 2 active beads in the upper portion of row *n,* to 1 active bead in the lower portion of row *n*+1.

On an Arabic abacus you replace 10 active beads in row *n,* by 1 active bead in row *n*+1. Note that on an Arabic abacus entering $a^i_n \ldots a^k_0$ requires pushing $a^i_n + \ldots + a^k_0$ beads. Entering $a^i_n \ldots a^k_0 + b^i_m \ldots b^k_0$ requires entering $a^i_n + \ldots + a^k_0 + b^i_m + \ldots + b^k_0$ beads, plus carrying. Doing so would be an exceedingly tedious process, a major reason why the abacus is not used in cultures with Arabic numerals. In the seventeenth century, however, Blaise Pascal invented a mechanical "adding machine" (called the *pascaline*) that is essentially a mechanized Arabic abacus, which performs "carrying" automatically.[8]

4. Multiplication

---

[6] For an online working abacus, in various number bases including Chinese and Roman, go to: http://www.tux.org/~bagleyd/java/AbacusAppJS.html

[7] Image from: http://www.hh.schule.de/metalltechnik-didaktik/users/luetjens/abakus/rom-abakus-en.htm

[8] See, for example, http://www.macs.hw.ac.uk/~greg/calculators/pascal/Pascaline_Frames.htm.

The interpretation of the 2-place ***multiplication operator*** $\times$ is defined as follows:

For any Arabic/Roman numeral,
$\Im(a^i_n \ldots a^k_0 \times b^i_m \ldots b^k_0) = \Im(a^i_n \ldots a^k_0) \bullet \Im(b^i_m \ldots b^k_0)$

Like the familiar long multiplication procedure with Arabic numerals, in the multiplication algorithm for multiplying with Roman numerals is grounded *law of distribution* of addition over multiplication from number theory. In law may be formulated is various equivalent ways:

Law of Distribution

$(x+y) \bullet (u+v) = [(x\bullet u)+(x\bullet v)]+[(y\bullet u)+(y\bullet v)]$

$(x_1+\ldots+x_n) \bullet (y_1+\ldots+y_m) = [(x_1\bullet y_1)+\ldots+(x_1\bullet y_m)]+\ldots+[(x_n\bullet y_1)+\ldots+(x_n\bullet y_m)]$

$x\bullet y = \Sigma^n_{i=1} x_i \bullet \Sigma^m_{j=1} y_j = \Sigma^n_{i=1}\Sigma^m_{j=1} x_i \bullet y_j$

We now make use of the fact that a number in Arabic and Roman notation $e_n \ldots e_0$ may be formulated $(1^n\, e_n)+\ldots+(1^0\, e_0)$. That is,

$\Im(e_n \ldots e_0) = (\mathbf{1}^n \bullet \Im(e_n))+\ldots+(\mathbf{1}^0 \bullet \Im(e_0))$

It then follows that binary multiplication obeys the following distribution law:

Theorem. For any Arabic/Roman numerals $a_n \ldots a_0$ and $b_m \ldots b_0$,

$\Im(a_n \ldots a_0 \times b_m \ldots b_0) =$

$[(\mathbf{1}^n \bullet \Im(a_n))+\ldots+(\mathbf{1}^0 \bullet \Im(a_0))] \bullet [(\mathbf{1}^n \bullet \Im(b_m))+\ldots+(\mathbf{1}^0 \bullet \Im(b_0))] =$

$[((\mathbf{1}^n \bullet \Im(a_n)) \bullet (1^n \bullet \Im(b_m))+\ldots+((\mathbf{1}^n \bullet \Im(a_n)) \bullet (\mathbf{1}^0 \bullet \Im(b_0)))] + \ldots +$
$[((\mathbf{1}^0 \bullet \Im(a_0)) \bullet (\mathbf{1}^n \bullet \Im(b_m))+\ldots+((\mathbf{1}^0 \bullet \Im(a_0)) \bullet (\mathbf{1}^0 \bullet \Im(b_0)))]$

It is also possible to state the results of multiplying one basic numeral by another. Indeed this is essentially what we learned in elementary school when we memorized (or failed to) the multiplication table.

Theorem (Multiplication Table for Basic Arabic Numerals). If x and y are in {1,2,3,4,5,6,7,8,9} and x^n and y^m are basic Arabic numerals, then $x^n \times y^m$ is as indicated in the table below:

$\times$	1^m	2^m	3^m	4^m	5^m	6^m	7^m	8^m	9^m
1^n	1^{n+m}	2^{n+m}	3^{n+m}	4^{n+m}	5^{n+m}	6^{n+m}	7^{n+m}	8^{n+m}	9^{n+m}
2^n	2^{n+m}	4^{n+m}	6^{n+m}	8^{n+m}	1^{n+m+1}	$1^{n+m+1}2^{n+m}$	$1^{n+m+1}4^{n+m}$	$1^{n+m+1}6^{n+m}$	$1^{n+m+1}8^{n+m}$
3^n	3^{n+m}	6^{n+m}	9^{n+m}	$1^{n+m+1}2^{n+m}$	$1^{n+m+1}5^{n+m}$	$1^{n+m+1}8^{n+m}$	$2^{n+m+1}1^{n+m}$	$2^{n+m+1}4^{n+m}$	$2^{n+m+1}7^{n+m}$
4^n	4^{n+m}	8^{n+m}	$1^{n+m+1}2^{n+m}$	$1^{n+m+1}6^{n+m}$	2^{n+m+1}	$2^{n+m+1}4^{n+m}$	$2^{n+m+1}8^{n+m}$	$3^{n+m+1}2^{n+m}$	$3^{n+m+1}6^{n+m}$
5^n	5^{n+m}	1^{n+m+1}	$1^{n+m+1}5^{n+m}$	2^{n+m+1}	$2^{n+m+1}5^{n+m}$	3^{n+m+1}	$3^{n+m+1}5^{n+m}$	4^{n+m+1}	$4^{n+m+1}5^{n+m}$
6^n	6^{n+m}	$1^{n+m+1}2^{n+m}$	$1^{n+m+1}8^{n+m}$	$2^{n+m+1}4^{n+m}$	3^{n+m+1}	$3^{n+m+1}6^{n+m}$	$4^{n+m+1}2^{n+m}$	$4^{n+m+1}8^{n+m}$	$5^{n+m+1}4^{n+m}$
7^n	7^{n+m}	$1^{n+m+1}4^{n+m}$	$2^{n+m+1}1^{n+m}$	$2^{n+m+1}8^{n+m}$	$3^{n+m+1}5^{n+m}$	$4^{n+m+1}2^{n+m}$	$4^{n+m+1}9^{n+m}$	$5^{n+m+1}6^{n+m}$	$6^{n+m+1}3^{n+m}$
8^n	8^{n+m}	$1^{n+m+1}6^{n+m}$	$2^{n+m+1}4^{n+m}$	$3^{n+m+1}2^{n+m}$	4^{n+m+1}	$4^{n+m+1}8^{n+m}$	$5^{n+m+1}6^{n+m}$	$6^{n+m+1}4^{n+m}$	$7^{n+m+1}2^{n+m}$
9^n	9^{n+m}	$1^{n+m+1}8^{n+m}$	$2^{n+m+1}7^{n+m}$	$3^{n+m+1}6^{n+m}$	$4^{n+m+1}5^{n+m}$	$5^{n+m+1}4^{n+m}$	$6^{n+m+1}3^{n+m}$	$7^{n+m+1}2^{n+m}$	$8^{n+m+1}1^{n+m}$

By combining the basic calculations of the table with the law of distribution it is possible to calculate the product of complex Arabic numerals:

Example

$$
\begin{aligned}
51\times 25 &= [(1^1 \times 5)+(1^0 \times 1)] \times [(1^1 \times 2)+(1^0 \times 5)] \\
&= [(50 + 1)] \times [(20 + 5)] \\
&= [(50\times 20)+(1\times 20)] + [(50\times 5)+(1\times 5)] \\
&= 1000+20+250+5 \\
&= 1275
\end{aligned}
$$

It is essentially this process that we employ in the usual algorithm of "long multiplication:"

$$
\begin{array}{r}
25 \\
\underline{\times 51} \\
5 \\
20 \\
250 \\
\underline{+1000} \\
1275
\end{array}
$$

Thirty years after Pascal invented his mechanical abacus that allowed for addition by automatically converting ten units in column n to one unit in column n+1 (thereby carrying a unit values to the next higher order of ten), Leibniz invented a machine that would multiply by making use of the same distribution theorem used in long multiplication.[9] His *machine a calculer* is constructed so that by (1) situating an "adder crank" at a column representing a number *l,* (2) setting a selector value at *m,* and (3) turning the crank *n* times, the number *m* is automatically added to the number *l* as many times as the crank is turned, i.e. *m* times*,* with any 10 units at a column being converted ("carried") to one unit at the column to its immediate left. For example, to multiply 24 by 56, you first manually add 4 to 24 six times, by situating the "adder" crank at the column representing 4, setting the crank to the value 6 and then turning a crank 6 times, while the machine automatically carries as required. Then, displacing the adder crank one column to the right, you add 2 units (of 10's) to figure that now occupies that column by setting the crank to the value 2 and turning the crank 6 times, the machine automatically carrying as required. At that point 25 has been multiplied by 6. Then, displacing the adder crank to the 10's column, you manually add 4 to that column five times by setting the crank to the value 4 and cranking five times. Then, displacing the adder crank one column to the left again (to the 100's column), you then set the crank to the value 2 and crank five times. The product has been calculated by distribution:

$$(6\times4)+(4\times20)+(50\times4)+(50\times20)=1344$$

A version of the same algorithm works for Roman numerals, and can be implemented on the abacus. We make use of the same distribution law, but calculate a multiplication table for basic Roman numerals. Since there are only two, the table is much shorter, making life much happier for Roman children leaning how to multiply!

Theorem (Multiplication Table for Basic Roman Numerals). If *x* and *y* are in {I, V} and x^n and y^m are basic Roman numerals, then $x^n\times y^m$, then there are four cases:

$$\mathrm{I}^n\times\mathrm{I}^m = \mathrm{I}^{n+m}$$
$$\mathrm{I}^n\times\mathrm{V}^m = \mathrm{V}^m\times\mathrm{I}^n = \mathrm{V}^{n+m}$$
$$\mathrm{V}^n\times\mathrm{V}^m = \mathrm{I}^{n+m+1}\mathrm{I}^{n+m+1}\mathrm{V}^{n+m}$$

which may be summarized in the table below:

$\times$	I^m	V^m
I^n	I^{n+m}	V^{n+m}
V^n	V^{n+m}	$\mathrm{I}^{n+m1}\mathrm{I}^{n+m+1}\mathrm{V}^{n+m}$

[9] See http://www.physique.usherbrooke.ca/~afaribau/essai/essai.html

Example

$$
\begin{aligned}
LI \times XXV &= (V^1 I^0) \times (I^1 I^1 V^0)\\
&= (V^1\times I^1)(V^1\times I^1)(V^1\times V^0)(I^0\times I^1)(I^0\times I^1)(I^0\times V^0)\\
&= V^{1+1}V^{1+1}I^{1+0+1}I^{1+0+1}V^{1+0}I^{0+1}I^{0+1}V^0\\
&= V^2V^2\ I^2I^2V^1I^1I^1V^0\\
&= \cancel{DD}CCLXXV\\
&= MCCLXXV
\end{aligned}
$$

It may be helpful to display the steps in the familiar format of “long multiplication:”

$$
\begin{array}{r@{\qquad\qquad}r}
I^1I^1V^0 & XXV\\
\underline{\times V^1I^0} & \underline{\times LI}\\
V^0 & V\\
I^1 & X\\
I^1 & X\\
V^1 & L\\
I^2 & C\\
I^2 & C\\
V^2 & D\\
\underline{+V^2} & \underline{+D}\ .\\
\cancel{V^2V^2}I^2I^2V^1I^1I^1V^0 & \cancel{DD}CCLXXV\\
I^3I^2I^2V^1I^1I^1V^0 & MCCLXXV
\end{array}
$$

Multiplication on the Abacus

Multiplication on the abacus exactly follows the “long division” format for Roman numerals above. First the multiplicans and multiplicandum, expressed as Roman numerals, are enter on the abacus. Ideally the abacus has enough rows to do so while still leaving enough room to allow their product to be entered as it is calculated. Usually the multiplicans and multiplicandum are entered on the left with one or more empty row left between them to tell them apart, and the product is entered on the far right, again with unused rows separating it from the other numbers. When a complex numeral is entered, one bead is placed in the active position for each of its component basic numerals. The multiplication process consists of taking each bead of the multiplicans (i.e. each basic numeral), multiplying it by each bead of the multiplicandum (i.e. each of its basic numerals). As each of these partial products is calculated it is entered on that portion of the abacus reserved for representing the product. As new products are entered on top of the previous total the product grows. The abacus thus keeps running total of the component multiplications as the process progresses. In adding a new product to the total of previous products it may well be necessary to “carry,” as in normal addition. In practice, each time you finish multiplying the multiplicandum by a component bead in the multiplicans, you remove that bead from its active position. That way the total multiplicans diminishes as the total product grows. When all the beads of the multiplicans are gone, the process is completed.

Multiplying by I is easy. To multiply a bead x^n (a basic numeral) by 1^m you just enter an x bead on the $n+m^{th}$ row of the product. That is, you enter x^{n+m}. Indeed, multiplying the entire multiplicandum $e^i{}_n \ldots e^k{}_0$ by 1^m is easy. You just replicate (copy) the entire number in the product. In doing so, however, you must be sure to shift it to the left so that it terminates on row m. That is, you enter $e^{i+m}{}_n \ldots e^{k+m}{}_0$. Again, this will normally require carrying.

Multiplying by a bead by V takes more care. If you are multiplying a bead x^n (a basic numeral) by V^m, there are two cases. If x^n is I^n again, the process is easy. You simply enter V on the $n+m^{th}$ row, That is, you enter V^{n+m}. You may have to carry. If x^n is V^n, then you must enter $I^{n+m+1}I^{n+m+1}V^{n+m}$. Here because there are more numerals involved carrying is more likely.

The displayed "long multiplication" above may actually be read as a picture of the beads on the abacus. A basic numeral with superscript n represents a bead in row n, and I represents a unit bead (in the lower part) and V a five bead in the upper part. Each horizontal rank in the sum records the results of multiplying one bead of the multiplicans by one bead of he multiplicandum. As stipulated in the distribution law, the total product is then the sum of all this individual multiplications, i.e. the sum of all the ranks.

Representing $n \bullet m$ on a Roman abacus does not actually require many beads because sets of five units of a power are abbreviated by a five bead of that power. On an Arabic abacus, however, a complex number $a^i{}_n \ldots a^k{}_0$ is made up solely of unit beads. Hence multiplying using the distribution law would require multiple steps of multiplying by 1. That is, multiplying $a^i{}_n \ldots a^k{}_0$ by $b^k{}_0, \ldots, b^i$ would require $b^k{}_0, \ldots, b^i$ steps of entering the complex numeral $a^i{}_n \ldots a^k{}_0$. to a running total. Though feasible (a machine could do it) the process it would be very tedious, another reason why the abacus is not used in cultures that use Arabic numerals.

4. Division

The interpretation of the 2-place ***division operator*** / is defined as follows:

For any Arabic/Roman numerals $a^i{}_n \ldots a^k{}_0$ and $b^i{}_m \ldots b^k{}_0$ such that $\Im(b^i{}_m \ldots b^k{}_0) \leq \Im(a^i{}_n \ldots a^k{}_0)$, $\Im(a^i{}_n \ldots a^k{}_0 \ / \ b^i{}_m \ldots b^k{}_0)$ with remainder r is defined the greatest x such that $\Im(b^i{}_m \ldots b^k{}_0) \bullet x = \Im(a^i{}_n \ldots a^k{}_0) + r$.

There is a straightforward way to calculate $a^i{}_n \ldots a^k{}_0 \ / \ b^i{}_m \ldots b^k{}_0$ and r.

The algorithm consists of repeated applications of the following truth of arithmetic:

$$(x + (y \bullet 10^n)) / y = 10^n + (x / y)$$

In the notation of Roman numerals in which the simple concatenation of basic numerals is equivalent to addition the law may be formulated:

$$(x^i_n \ldots x^k_0\, y^{k+c}_m \ldots y^{j+c}_0) \,/\, (y^k_m \ldots y^j_0) \;=\; \mathrm{I}^c \;\; (x^i_n \ldots x^k_0) \,/\, (y^k_m \ldots y^j_0)$$

To calculate $a^i_n \ldots a^k_0 \,/\, b^i_m \ldots b^k_0$ and r, first $a^i_n \ldots a^k_0$ is reformulated by expansion so that it is equivalent to some $c^i_n \ldots c^k_0\, b^{k+c}_m \ldots b^{i+c}_0$. It is possible to do so because is greater than $b^i_m \ldots b^k_0 \leq a^i_n \ldots a^k_0$. Add I^c to the quotient. If $c^i_n \ldots c^k_0 < b^i_m \ldots b^k_0$, stop, and set $r = c^i_n \ldots c^k_0$. If $b^i_m \ldots b^k_0 \leq c^i_n \ldots c^k_0$, the process is repeated on the new diminished dividendum. That is, $c^i_n \ldots c^k_0 \,/\, b^i_m \ldots b^k_0$ is calculated, obtaining some $d^u_s \ldots d^v_0 b^{i+t}_m \ldots b^{k+t}_0$ equal to $c^i_n \ldots c^k$, adding I^t to the quotient. The $d^u_s \ldots d^v_0 = r$ if $d^i_n \ldots d^k_0 < b^i_m \ldots b^k_0$ and the process stops, or $d^u_s \ldots d^v_0$ is the new diminished dividendum, and the process is repeated. Since at some point the process stops because the diminished dividendum keeps getting smaller.

Example

CCCLV / XXV =

$I^2I^2I^2V^1V^0$ / $I^1I^1V^0$ =

$I^2V^0I^2I^2V^1$ / $I^1I^1V^0$ =

I^2V^0 $I^{1+1}I^{1+1}V^{0+1}$ / $I^1I^1V^0$ =

I^1 + (I^2V^0 / $I^1I^1V^0$) =

I^1 + ($V^1V^1V^0$ / $I^1I^1V^0$) =

I^1 + ($V^1I^1I^1I^1I^1I^1V^0$ / $I^1I^1V^0$) =

I^1 + ($V^1I^1I^1I^1$ $I^1I^1V^0$ / $I^1I^1V^0$) =

I^1 + I^0 + ($V^1I^1I^1I^1$ / $I^1I^1V^0$) =

I^1 + I^0 + ($V^1I^1I^1V^0V^0$ / $I^1I^1V^0$) =

I^1 + I^0 + (V^1V^0 $I^1I^1V^0$ / $I^1I^1V^0$) =

I^1 + I^0 + I^0 + (V^1V^0 / $I^1I^1V^0$) =

I^1 + I^0 + I^0 + ($I^1I^1I^1$ $I^1I^1V^0$ / $I^1I^1V^0$) =

I^1 + I^0 +I^0 + I^0 + ($I^1I^1I^1$ / $I^1I^1V^0$) =

I^1 + I^0 +I^0 + I^0 + ($I^1I^1V^0V^0$ / $I^1I^1V^0$) =

I^1 + I^0 +I^0 + I^0 + (V^0 $I^1I^1V^0$ / $I^1I^1V^0$) =

I^1 + I^0 + I^0 +I^0 + I^0 + (V^0 / $I^1I^1V^0$) =

XIIII, with remainder V

Note that if the numeral system was extended to allow for the representation of decimal factions by allowing basic numerals for the negative posers of 10, then the quotient could be a calculated to any desired degree of precision.

Division on the Abacus

Like multiplication "long division" on the abacus begins by entering both dividens and the dividendum, leaving room for the quotient. The entire dividens is "divided into" the dividendum as indicated in the algorithm. The new diminished dividendum is exhibited instantly by subtracting the relevant power of the dividens directly from the old dividendum. As it is subtracted a I of the relevant power is added to the quotient. When the dividendum becomes smaller than the dividens, the process stops and the remaining dividendum is literally "the remainder". The process is easy – easier than multiplication, and easier than either multiplication or division with Arabic numerals.