Was John Napier the First Modern Computer Scientist?

Joint Meeting
The History and Pedagogy of Mathematics (HPM), Americas Section and
The Midwest History of Mathematics Conference (MHMC)
Wabash College, Crawfordsville, IN
October 18, 2014

Patricia Baggett Andrzej Ehrenfeucht
Dept. of Math Sci. Computer Science Dept.
New Mexico State Univ. University of Colorado
Las Cruces, NM 88003-8001 Boulder, CO 80309-0430 baggett@nmsu.edu
andrzej@cs.colorado.edu

http://www.math.nmsu.edu/~breakingaway/

Outline

1. A little history: A universal computing device
2. John Napier’s overlooked idea (1617)
   - Location numbers on a chessboard
   - Why location numbers were ignored
   - Martin Gardner (1986)
3. Napier’s abacus as a teaching tool
   - Decimal boards
   - Possible applications in today’s schools
4. Final remarks

1. A little history: A universal computing device

The history of data processing can be looked at from two different points of view, *engineering* and *mathematical*.

An engineering point of view covers the *development of data storage* from tally sticks, knots on cords, and clay tablets, to modern memory sticks.
And the evolution of computing tools starts with fingers, beans, and pebbles, and currently includes iphones, computers, and other electronic devices.

A mathematical point of view deals with the *development of the underlying ideas*, independent of whether they were abandoned and forgotten, or whether they had great practical significance.

A discussion of Napier's achievement, which provided logarithmic tables, the slide rule, and to a lesser degree, "Napier's bones", belongs to the engineering view of history.

But his "location numbers" and "computation on a chess board", which never had any application, belong to a mathematical view of the history of computation, and they are the subject of this talk.

Computer science was created around the middle of the twentieth century and is based on two abstract concepts: *data* and *algorithms*.

The abstract concept of data is a part of “information theory”, created by Claude Shannon (1916-2001), together with several related concepts such as the now familiar measure of the capacity of information storage devices in kilobytes or megabytes.

The concept of algorithm was created, under several different names, by David Hilbert (1862-1943) and his followers during the first quarter of the twentieth century, as part of his attempt to clarify the philosophical foundations of mathematics.

An algorithm is a detailed, step-by-step description of a data processing procedure. We still use many mathematical algorithms that were designed more than two thousand years ago.

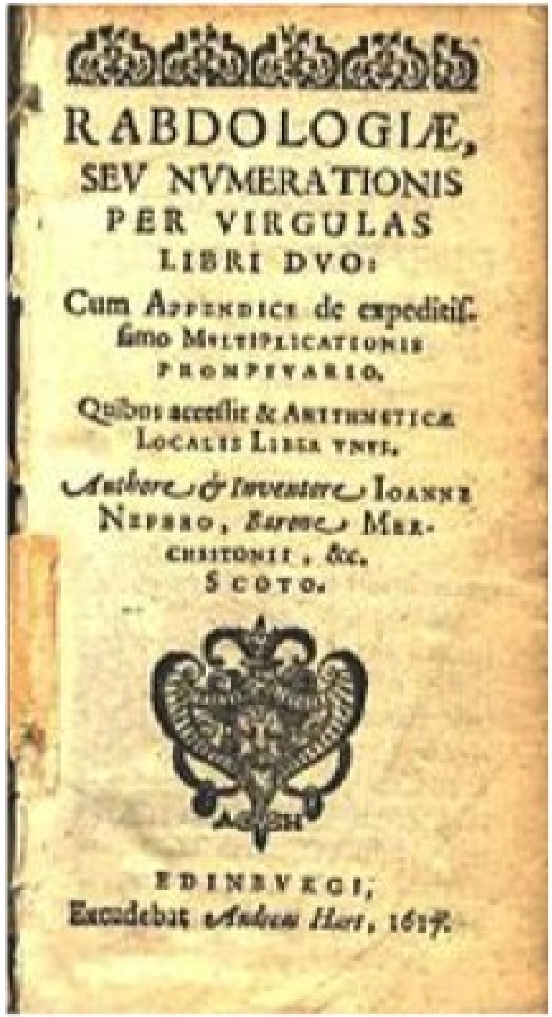
But no systematic study of algorithms, and no reasonable definition of them, were known until the work of Hilbert and his followers.

But the key insight that made all the practical difference belongs to Alan Turing (1912-1954), who showed that it is possible to construct one device that is capable of carrying out any algorithm, regardless of how complex and difficult it may be.
The only resource that such a “universal computer” needs is access to sufficiently large data storage.
Such universal computers are now small, cheap, and mass-produced, and they are the “brains” of all “smart” electronic devices.

But was Alan Turing the first person who thought that it is possible to construct a universal computer?
We think that John Napier (1550-1617) attempted to create such a device, and that he even might have believed that his chessboard, filled with location numbers, was a universal device.

(It was not, but it is still a very powerful computing device.)

2. John Napier's overlooked idea from his Rabdologiae (1617)


RABDOLOGIÆ,
SEV NVMERATIONIS
PER VIRGULAS
LIBRI DVO:
Cum APPENDICE de expeditissimo MVLTIPLICATIONIS
PROMPTVARIO.
Quibus accessit & ARITHMETICÆ
LOCALIS LIBER VNVS.
Authore & Inventore IOANNE
NEPERO, Barone MERCHISTONII, &c.
SCOTO.

EDINBVRGI,
Excudebat Andreas Hart, 1617.

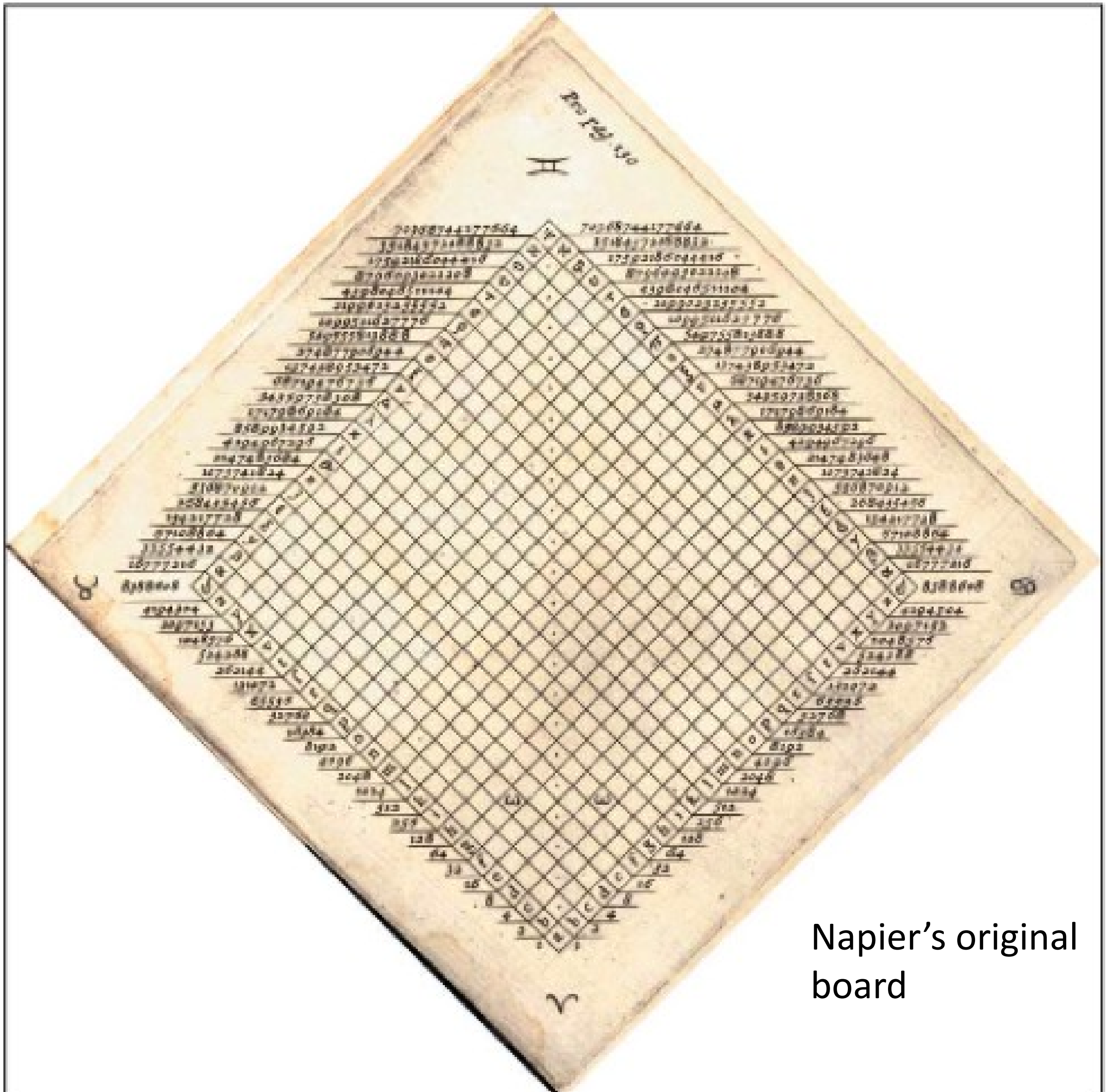
Location numbers on a chessboard

This is the last topic that Napier describes in *Rabdologiae* (1617), introducing it as follows:

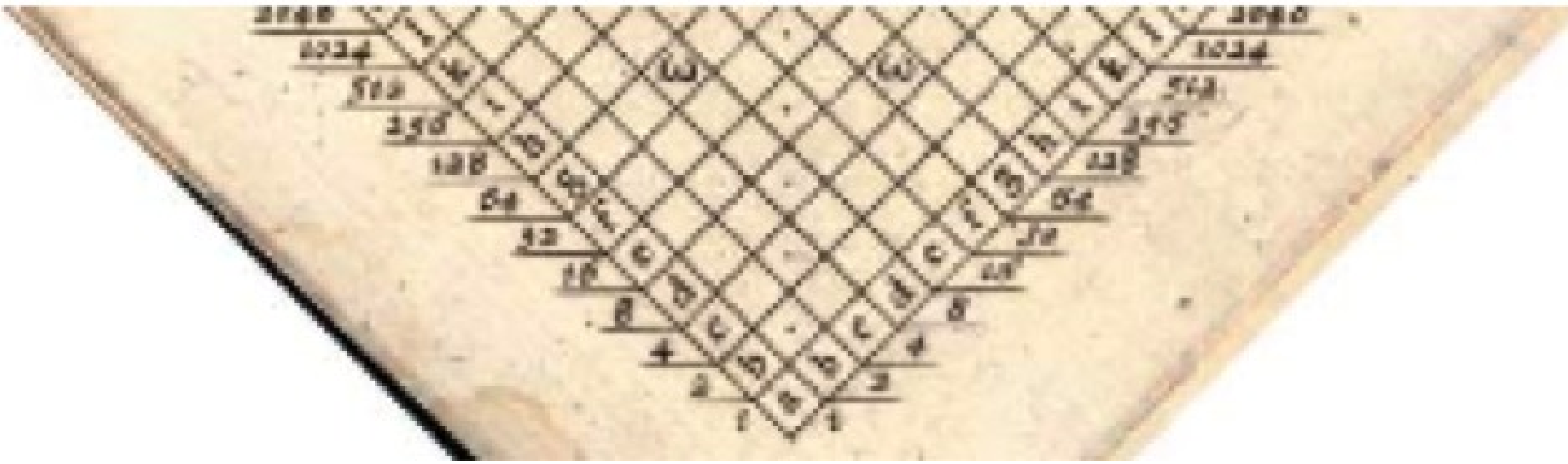
While working in my spare time on these short methods and seeking ways in which the tedium of calculations can be removed, I developed not only my logarithms, my rabdology (rods), my promptuary for multiplication, and other things but also a method of arithmetic on a flat surface. ***As it performs all the more difficult operations of common arithmetic on a chessboard****, it might be described as more a lark than a labor, for it carries out addition, subtraction, multiplication, division, and (yes!) extraction of roots* ***purely by moving counters from place to place****. …*

But he thinks that his computing device will do more than arithmetic calculations. In describing the use of powers of two in its construction, he writes:

Let it be a rod …, divided into equal parts, one for each counter or number [power of two] you desire it to hold. If, then, you want it to hold 16 counters or 16 numbers, you will divide it into 16 parts. In which case the sixteenth number would be 32,768, and the rod will calculate all numbers less than 65,536, which is sufficiently large for ordinary use. … ***If however you want to work with larger numbers (such as sines, tangents and secants)*** *make a rod 48 fingers long and divide it into the same number of parts to accommodate 48 counters and 48 numbers, the last being 140,737,488,355,328.*



Napier's original board



2048
1024
512
256
128
64
32
16
8
4
2
1
2048
1024
512
128
64
4
1

2^n*2^m							2^m
			64	32	16	8	4
			32	16	8	4	2
2^n			16	8	4	2	1

Think of a chessboard of any size. Put the value 1 in the right bottom corner and let all the rows and all the columns contain numbers forming geometric progressions with a factor of 2. These are location numbers on a chessboard.

Any configuration of tokens on a board represents a number.

Each token has the value of its location, and we take the sum of values of all tokens. (Several tokens can be stacked up on one location.)

Tokens can be moved according to simple regrouping rules that do not change the total value on the board.

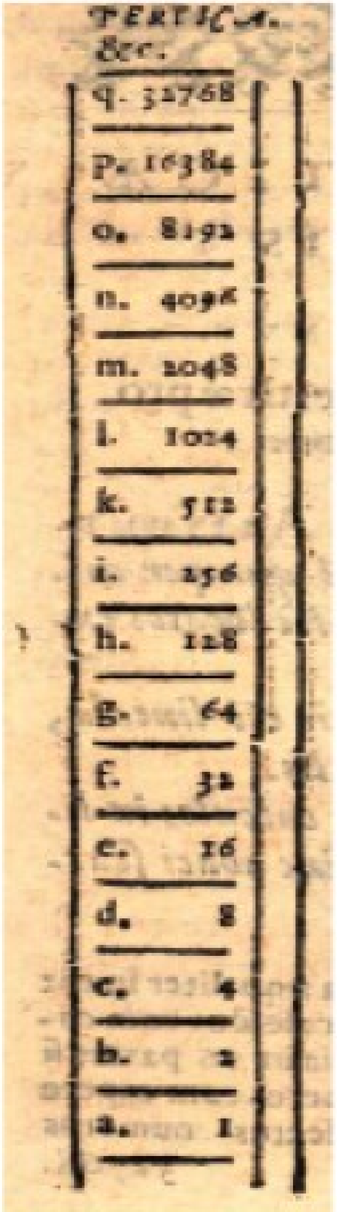
2^n*2^m							2^m
			64	32	16	8	4
			32	16	8	4	2
2^n			16	8	4	2	1

(1) A token can be moved along a diagonal of locations having the same values.
(2) A token can be replaced by two tokens either to its right or just below.
(3) Two tokens on the same location can be replaced by one token to the left or above.

Computations are performed as follows:

Tokens are put on the board, their configuration is simplified according to rules (1) through (3) above, and the result is recorded.

Different operations require different ways of putting tokens on a board, but the rules of regrouping are always the same.



We may say now that Napier represented numbers in base two.
But he did not have the concept of base 2.
Instead he created his own notation, where letters represented powers of two.

a = 1, b = 2, c = 4, d = 8, and so on …,
and values of the letters were added.

Thus dba = 8 + 2 + 1 = 11.

Napier described in detail how to translate decimal numbers into his notation, and how to translate the result of computations back into decimals.

He showed how to use his board to compute the values of all of the five operations he mentioned in the preface.

Comparing location numbers to Turing machines

Turing described his computing devices in these terms:

You have an infinite tape that is divided into squares.
Each square can hold one character from an alphabet.
Operations are erasing an existing character and replacing it by another, and moving left and right to the nearest location.
(It is enough to use an alphabet that contains only two characters, 1 and blank.)
Each device is described by a finite set of instructions that tell which operations to use.

The computational power of such devices depends on their instructions.
And only some of them are universal.
(Google: the smallest universal Turing machines, John Conway's Game of Life, and Stephen Wolfram's cellular automata)

Of course writing and erasing characters and putting on and removing tokens are equivalent actions, and a two-dimensional array is more convenient to use. So we can ask the question whether Napier's board is a universal computing device.

The answer is no; not all algorithms can be carried out by putting tokens on the board and simplifying the configuration.
But it is a very powerful computational tool, so Napier was right when he said, "it performs all more difficult operations of common arithmetic."

Why location numbers were ignored

This is a matter of speculation, but calculation with tokens (jetons) was already waning in the seventeenth century.
Financial institutions were using “modern” bookkeeping, introduced by Luca Pacioli (1445-1517), which required using written computation, and technical computations were done with the newly invented slide rule.
So computation on a chessboard could have been viewed as old-fashioned.

But Napier’s use of powers of two in his original but strange notation was probably the main reason that no one wanted to use his invention.

Paradoxically his counting board would be even better if it were made in base 10 and not base 2, because its computational power lies in using geometric progressions in rows and columns, which makes it similar to a two-dimensional slide rule, and not in using base 2.

Martin Gardner (1914-2010)

In 1986 Martin Gardner wrote an article, *Napier's Abacus*[1], in which he said,

"[Napier] described a curious method of calculating by moving tokens across a chessboard." And, "It is the world's first binary computer, and it came almost 100 years before Leibnitz explained how to calculate with binary numbers!"

1In *Knotted Doughnuts and Other Mathematical Entertainments*

3. Napier’s abacus as a teaching tool

Can a Napier’s abacus be used as a teaching tool, together with other manipulatives such as base-ten blocks, unifix cubes, and other concrete representations of numbers?

Its original version is unsuitable for the following reasons:

(1) The fact that it uses only binary representations of numbers rules out its use in early grades.

(2) A binary representation requires many digits.
In *Rabdology* Napier shows a 24 by 24 board, which allows us to multiply two 7-digit decimal numbers.

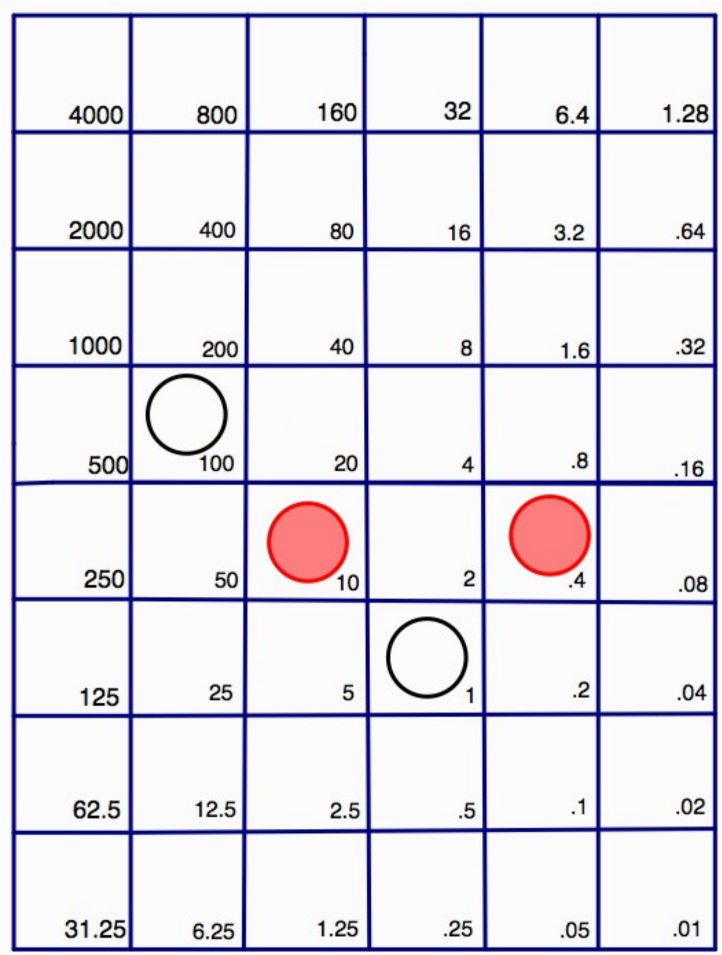
Assuming that each location is one square inch, the board would cover 4 square feet!

On a normal 8 by 8 chessboard, multiplication would be restricted to 2-digit decimals because the biggest location number on such a board is $4^{7 = 16,384.}$

But as we said, the use of binary numbers was not an essential feature, so modifying the original design to accommodate a decimal representation of numbers makes it appropriate for all grades, and it also significantly decreases the number of locations that are needed to represent a single number.

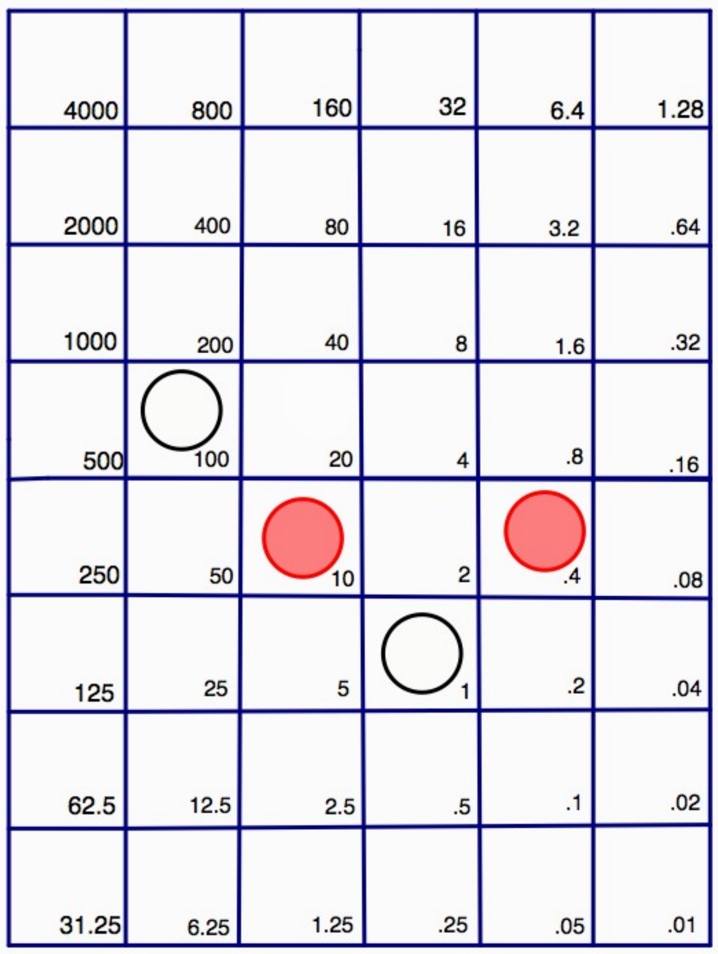
We'll call such a modification of Napier's abacus, a decimal board.

Decimal boards



On a decimal board, numbers in columns are still geometric progressions with the quotient 2. But numbers in rows are geometric progressions with the quotient 5. So on one of the two diagonals there is a geometric progression with the quotient 10. This allows us to represent numbers on any such a board in bases 2, 5, 10, and any other base b, which is a product of 2 and 5. Each decimal digit can be represented as one or two tokens on a board.

As before, exactly one location has the value 1. But the geometric progressions can extend to the right of 1, with values, .2, .04, .008, …, and below 1, with values, .5, .25, .125, … . Thus the decimal board allows us to represent not only whole numbers, but also decimal fractions.



Tokens used on a decimal board are two-sided. We use tokens that are white on one side and red on the other. The value of a white token on a board is the value of its location, but the value of a red token is the opposite of a white one, so it is a negative number. So all finite decimals, both positive and negative, can be represented.

When we compare the capacity of Napier's original abacus with a decimal board of the same size, we see that a decimal board has a bigger range and also allows us to represent more numbers and in more ways.

For example, compare two 8 by 8 boards:

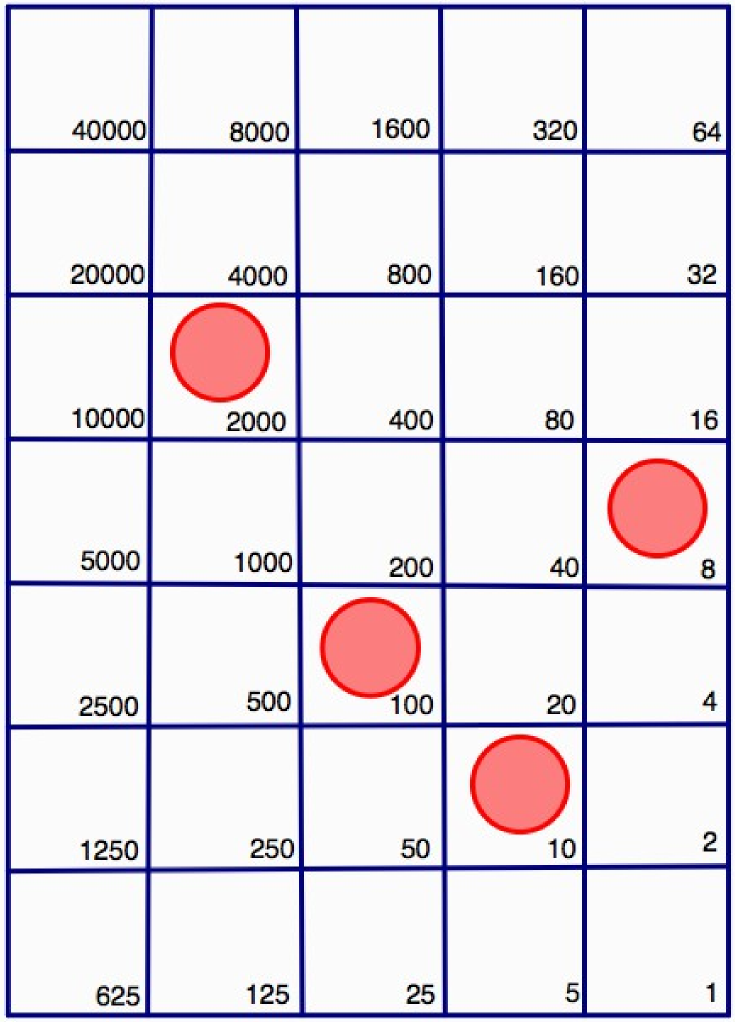
Both have 1 in their lowest right hand corners.

The range of numbers N represented by a single location:

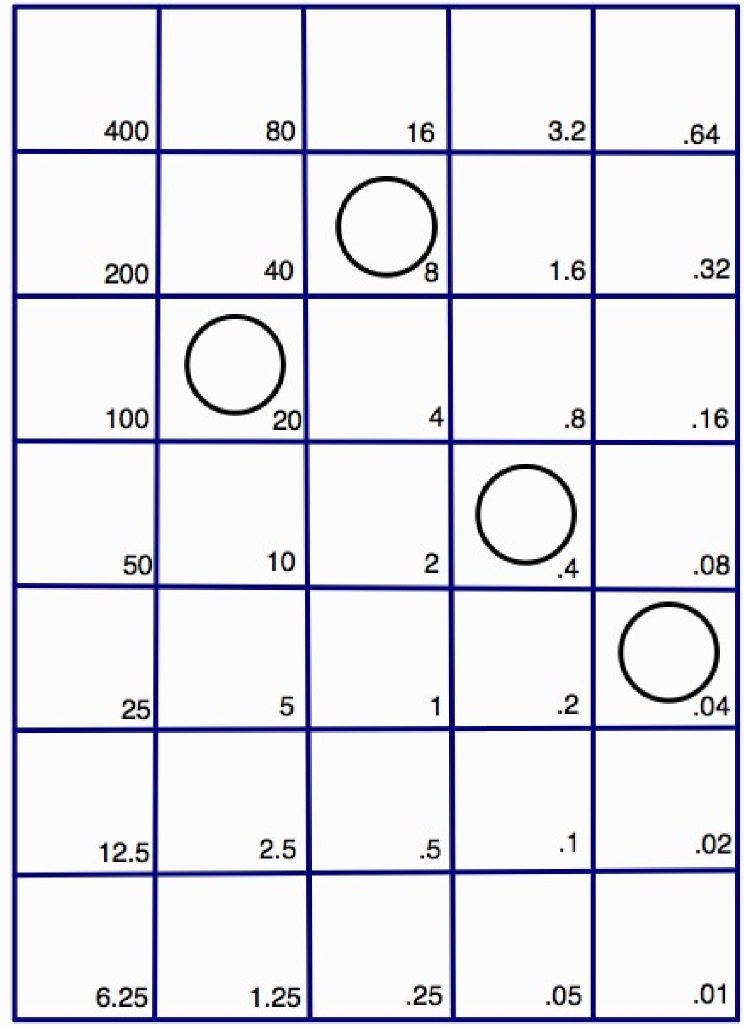
Napier's abacus	$1 \le N \le 16{,}384$
Decimal board	$-10{,}000{,}000 \le N \le 10{,}000{,}000$

The variety of representations:

Napier's abacus	base 2, for all numbers up to 32,767
Decimal board	base 10, for all numbers up to 99,999,999
	base 2, for numbers up to 255
	base 5, for numbers up to 390,624
	base 20, for numbers up to 3,199,999

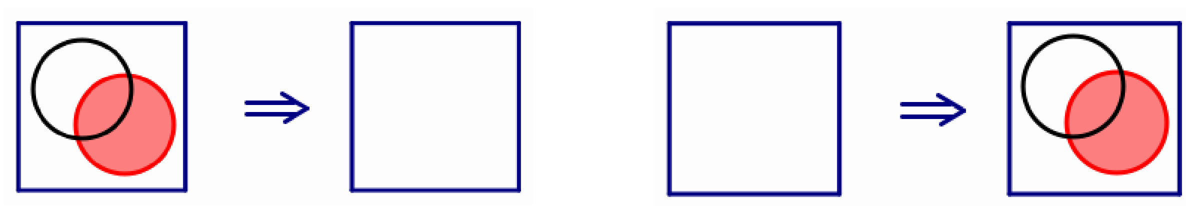

40000 8000 1600 320 64
20000 4000 800 160 32
10000 2000 400 80 16
5000 1000 200 40 8
2500 500 100 20 4
1250 250 50 10 2
625 125 25 5 1

-2118

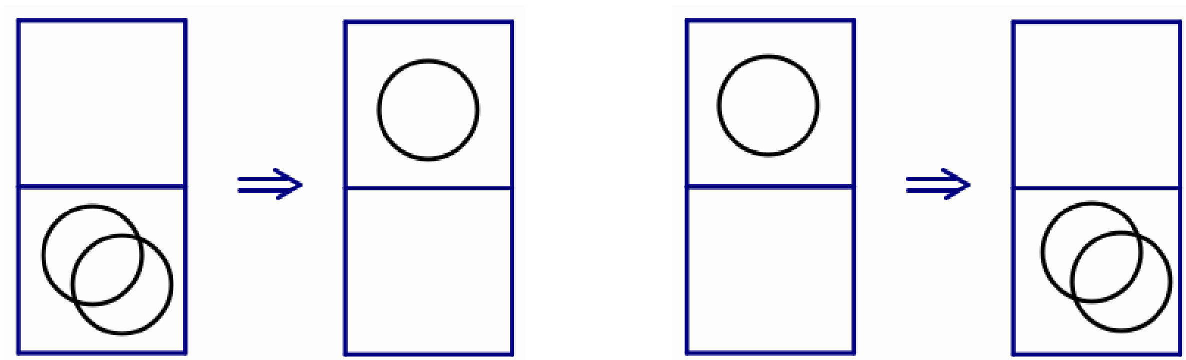

400 80 16 3.2 .64
200 40 8 1.6 .32
100 20 4 .8 .16
50 10 2 .4 .08
25 5 1 .2 .04
12.5 2.5 .5 .1 .02
6.25 1.25 .25 .05 .01

28.44

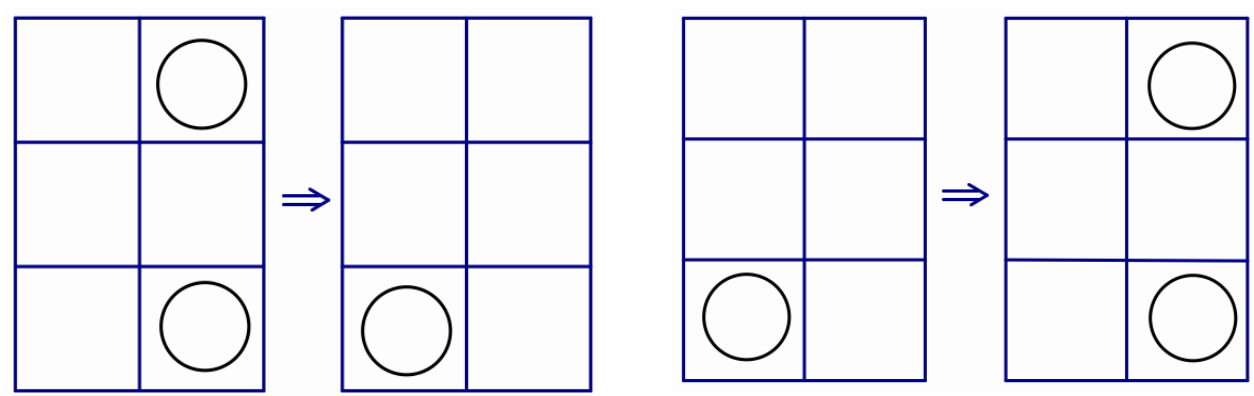
Six rules of moving tokens are sufficient to transform any configuration on a board to any other configuration representing the same number:



Two rules corresponding to the arithmetic equality -1+1=0



Two rules corresponding to the arithmetic equality 1 + 1 = 2



Two rules corresponding to the arithmetic equality 1 + 4 = 5

Applications of decimal boards

There is a consensus that knowing arithmetic is important for all students, and that it is a prerequisite for learning other mathematics.
But what it means to "know arithmetic" is far from being clear.
There are strong disagreements about the role of computational skills, the knowledge of algebraic properties of operations, the knowledge of specific facts, skills in applying mathematics, and the ability to solve difficult problems.

But it is well known that a person's ability to do mental arithmetic, without any external devices to store numbers, is very limited.
This is not a practical problem today, because even simple calculators provide more computing power than is needed in most situations.
But calculators don't solve the problem of teaching arithmetic.
They are "black boxes" that provide answers but hide the steps that lead to solutions.

So there is a need for low-tech computational devices that allow young students to perform fairly complex computations and that *make visible how each action that a student performs* influences the result.

In schools today the need for a low-tech device is met by paper-and-pencil computation, and the techniques that students learn are mostly the “standard” arithmetic algorithms that were used by accountants for several hundred years.

Readily available teaching aids such as base-10 blocks, unifix cubes, and Cuisinaire rods do not provide additional computing power; they only give a physical representation of mental arithmetic operations.

The arithmetic taught today still follows the pattern that was designed when the goal was to train human computers to perform arithmetical operations without any errors, quickly, and automatically.
And the sequence of learning arithmetic operations was determined by which new skill required the use of skills that were previously learned.
The same pattern is seen in modern classrooms.

Children start with rote counting; then they learn addition and subtraction by counting up and counting down.

They learn how to write and read numbers, and they memorize addition and subtraction facts.
And they practice written addition and subtraction.

They start learning multiplication as repeated addition;
they memorize multiplication facts, and then they learn a paper and pencil method.
“Long division”, which involves all previous skills, comes last.

This covers the arithmetic of whole numbers, which partially overlaps with three other topics: common fractions, decimal fractions, and negative numbers.

This way of teaching arithmetic has some drawbacks.

- The skills of written computation (especially multiplication and division) have minimal practical use.
  When we asked undergraduate majors in computer science how often they used written division outside school and college classrooms, more than half answered “never”. So a considerable amount of classroom time is spent on practicing useless skills.

- By learning skills sequentially, a student who doesn’t master one skill, never masters the remaining ones. It also leads to endless repetition. One teacher said, “Before I start fractions, I always review multiplication facts. But when students don’t know their addition facts, I just give them calculators.”

- Written algorithms are rigid. It is not easy to erase part of one’s work and do it differently. This gives the wrong impression that there is one right way to do a computation, and makes experimentation difficult.

Use of a decimal board as a platform for teaching arithmetic

Disclaimer:

We don’t say that written computations should not be taught, or that memorizing multiplication facts is not needed.

We think that written computation should be taught as one of the possible methods.
And most college students whom we asked thought that memorizing multiplication facts is an important part of learning arithmetic.

But we think that *written computation should not be a platform for teaching arithmetic*.

When one uses a decimal board:

1. The topics of whole numbers, decimals, and negative numbers can be taught at the same time, because any algorithm that is implemented on this board can be applied simultaneously to all these numbers.

2. Operations can be taught in any order. For example, neither multiplication nor subtraction is a prerequisite for division. This is the case because the rules for moving tokens on a board do not depend on what operation is being implemented.

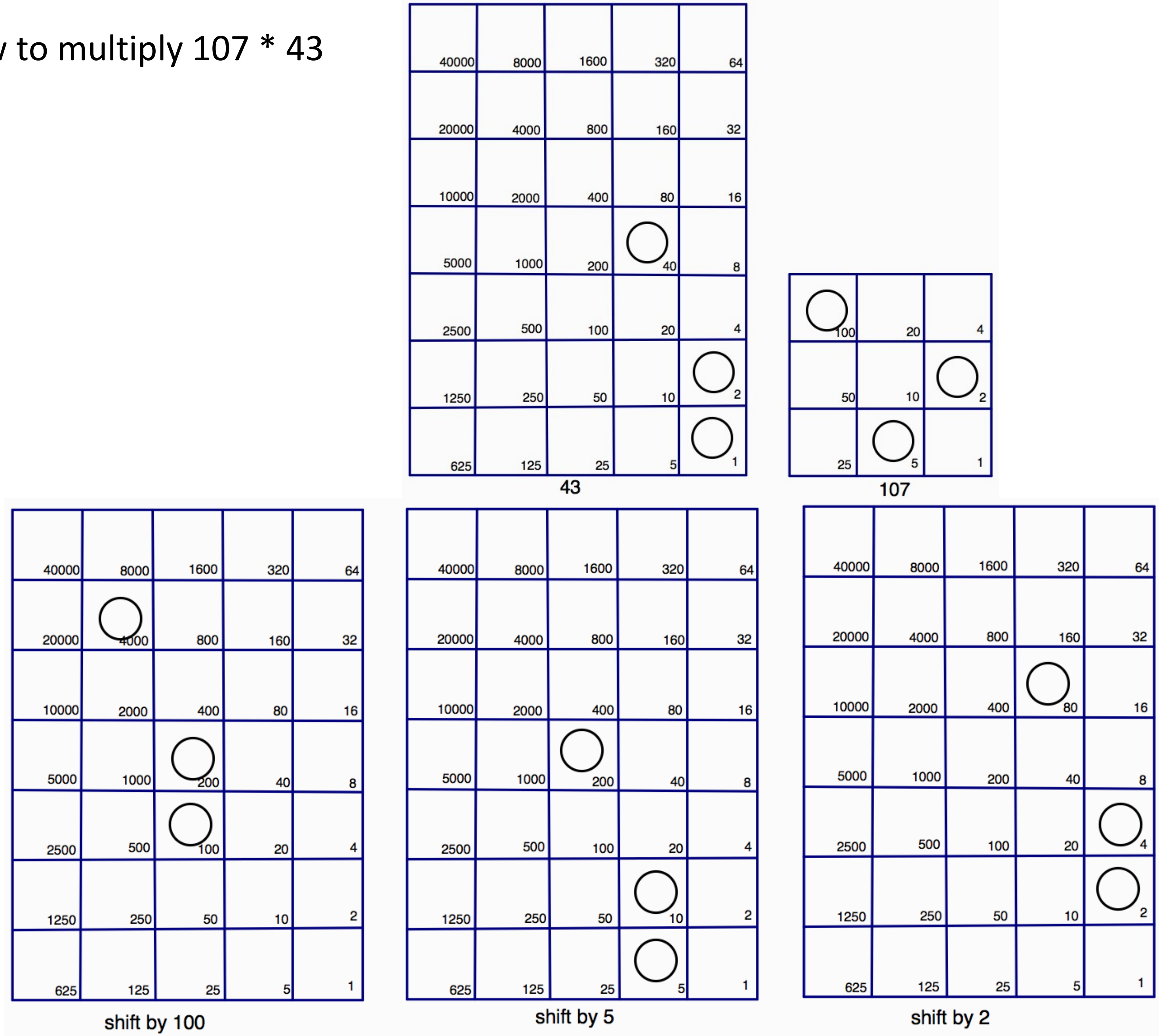
3. Skills in mental computation, together with the automatic recall of “arithmetic facts”, facilitate, and significantly speed up, the use of a decimal board, but they are not a prerequisite for learning any algorithm.

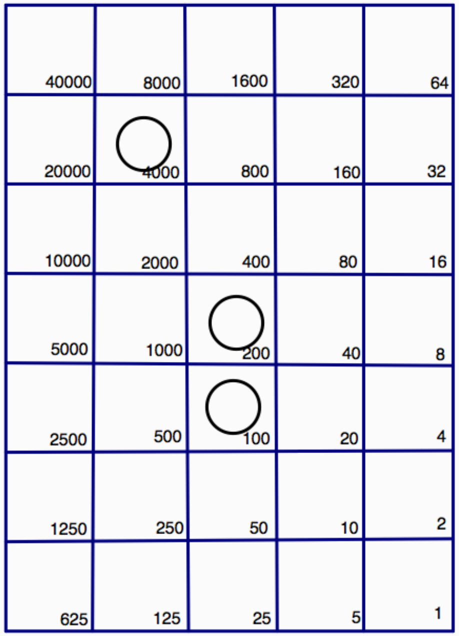
4. Computation is flexible; retracting a step and doing something different is easy on a board, and this encourages experimentation.

5. Experimenting is safe, because changing one’s method of regrouping, as long as it is consistent with the six basic rules of moving tokens, never creates an error.

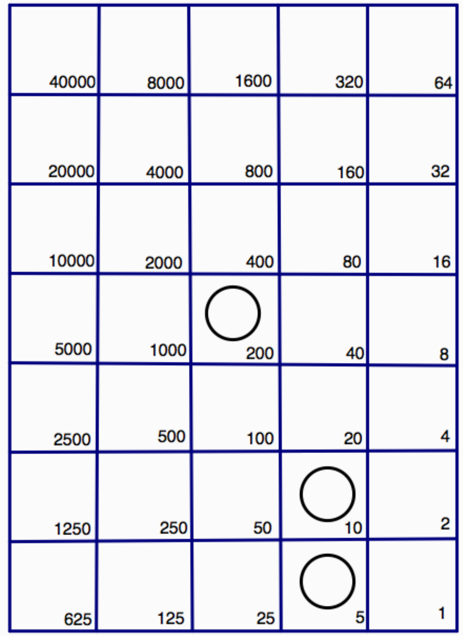
Thus the decimal board seems to be a better platform for teaching arithmetic than the platform of written computation, which is used currently.

How to multiply 107 * 43

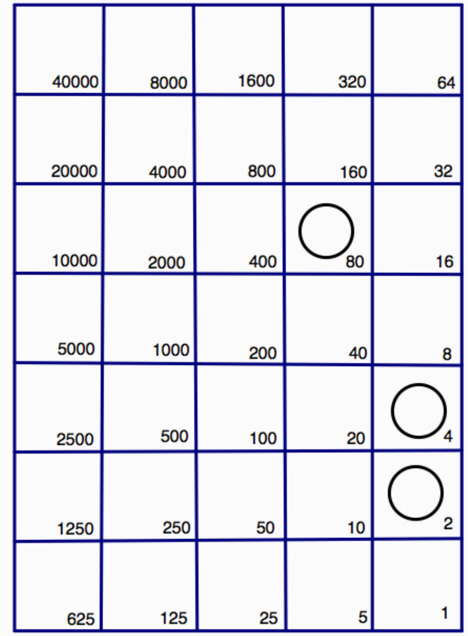




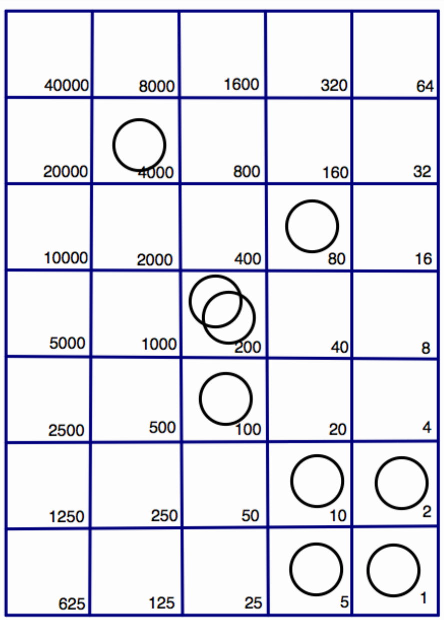
shift by 100



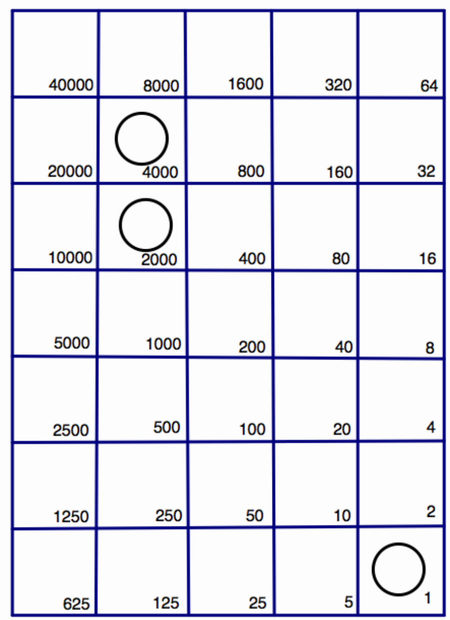
shift by 5



shift by 2



combining the three shifts



simplyfying to get the answer 4601

Final Remarks

Today many topics in the history of mathematics are undergoing revision. In addition to having better access to historical sources, we have changed our view about the nature of mathematics during the last one and a half centuries.

The creation of new domains of mathematics has made obsolete the old classification schema that divides mathematical topics into arithmetic, algebra, geometry, and calculus.

When Napier's location numbers were viewed as a (rather clumsy) attempt to use binary notation, they did not garner much attention.

But if (as we think) they were an attempt to create a universal computing device, governed by a small set of simple rules, location numbers were an unprecedented idea that changed all data processing 300 years later.

Neither the set of rules that Napier presented, nor the rules for a decimal counting board that we have shown above, are universal.

But because they are sufficient to cover most of the computation methods that are taught in elementary and middle schools, they may become a good platform for teaching arithmetic.

Our experiments with using decimal boards in mathematics courses for teachers are very encouraging.

If you would like more details:

baggett@nmsu.edu
andrzej.ehrenfeucht@cs.colorado.edu

Thank you!