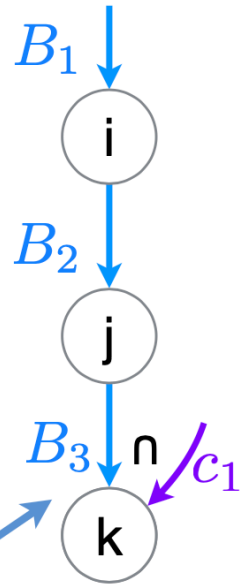
Notation

Pat Hanrahan

"Semantics, Not Syntax"

$$A_{ij} = \sum_k B_{ijk} c_k$$




```
for (int i = 0; i < m; i++) {

  for (int pB2 = B2_pos[i]; pB2 < B2_pos[i+1]; pB2++) {
    int j = B2_crd[pB2];


    int pA2 = i*n + j;
    int pB3 = B3_pos[pB2];
    int pc1 = c1_pos[0];
    while (pB3 < B3_pos[pB2+1] && pc1 < c1_pos[1]) {
      int kB = B3_crd[pB3];
      int kc = c1_crd[pc1];
      int k = min(kB, kc);
      if (kB == k && kc == k) {
        A[pA2] += B[pB3] * c[pc1];
      }
      if (kB == k) pB3++;
      if (kc == k) pc1++;

    }
  }
}
```

Key operation is to coiterate over data structures

Intersection coiteration

Concrete index notation specifies order of computations and location of intermediate values

$$A_{ij} = B_{ij} + C_{ij} \longrightarrow \forall_i \forall_j \; A_{ij} = B_{ij} + C_{ij}$$

$$\alpha = \sum_i b_i c_i \longrightarrow \forall_i \; \alpha \mathrel{+}= b_i c_i$$

$$a_i = \sum_j B_{ij} c_j \longrightarrow \forall_i \forall_j \; a_i = t \textbf{ where } t \mathrel{+}= B_{ij} c_j$$

THE FOUNDATION OF THE GENERAL THEORY OF RELATIVITY

By A. EINSTEIN

"*Contraction*" *of a Mixed Tensor.*—From any mixed tensor we may form a tensor whose rank is less by two, by equating an index of covariant with one of contravariant character, and summing with respect to this index ("contraction"). Thus, for example, from the mixed tensor of the fourth rank $A_{\mu\nu}^{\sigma\tau}$, we obtain the mixed tensor of the second rank,

$$A_{\nu}^{\tau} = A_{\mu\nu}^{\mu\tau} \quad \left(= \sum_{\mu} A_{\mu\nu}^{\mu\tau} \right),$$

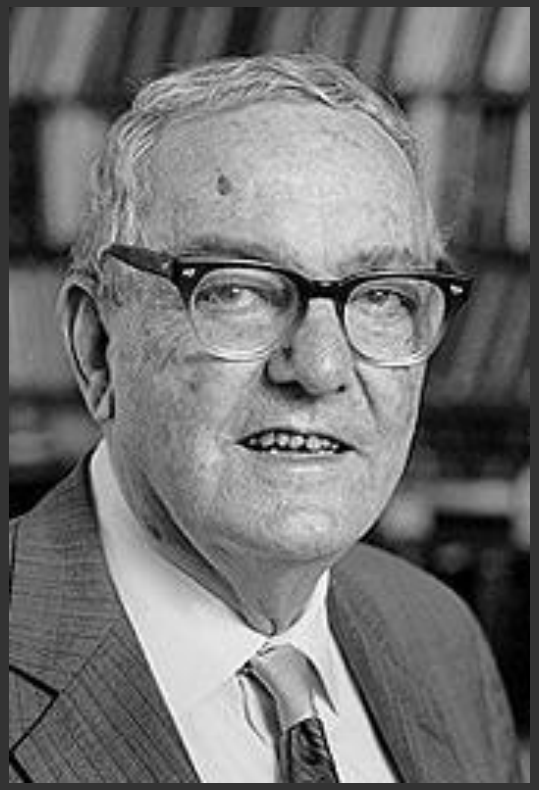
and from this, by a second contraction, the tensor of zero rank,

$$A = A_{\nu}^{\nu} = A_{\mu\nu}^{\mu\nu}.$$

The theory which is presented in the following pages conceivably constitutes the farthest-reaching generalization of a theory which, today, is generally called the
[1] "theory of relativity"; I will call the latter one—in order to distinguish it from the
[2] first named—the "special theory of relativity," which I assume to be known. The generalization of the theory of relativity has been facilitated considerably by Minkowski, a mathematician who was the first one to recognize the formal
[3] equivalence of space coordinates and the time coordinate, and utilized this in the construction of the theory. The mathematical tools that are necessary for general relativity were readily available in the "absolute differential calculus," which is based upon the research on non-Euclidean manifolds by Gauss, Riemann, and Christoffel, and which has been systematized by Ricci and Levi-Civita and has already been
[4] applied to problems of theoretical physics. In section B of the present paper I developed all the necessary mathematical tools—which cannot be assumed to be known to every physicist—and I tried to do it in as simple and transparent a manner as possible, so that a special study of the mathematical literature is not required for
[5] the understanding of the present paper. Finally, I want to acknowledge gratefully my friend, the mathematician Grossmann, whose help not only saved me the effort of
[6] studying the pertinent mathematical literature, but who also helped me in my search for the field equations of gravitation.

Representations

Herb Simon



Nobel Prize in Economics (1977)
“for his pioneering research into the decision-making process within economic organizations”

Turing Award (1975)
"basic contributions to artificial intelligence, the psychology of human cognition, and list processing"

Number Scrabble

Goal: Pick three numbers that sum to 15

A:

B:

Number Scrabble

Goal: Pick three numbers that sum to 15

A: 8 4 5

B: 2 3 ?

Number Scrabble

Goal: Pick three numbers that sum to 15

A: 8 4 5

B: 2 3 ?

Number Scrabble

Goal: Pick three numbers that sum to 15

A:

B:

Number Scrabble

Goal: Pick three numbers that sum to 15

A:

B:

Number Scrabble

Goal: Pick three numbers that sum to 15

A: 8 4

B: 2

Number Scrabble

Goal: Pick three numbers that sum to 15

A: 8 4

B: 2 3

Number Scrabble

Goal: Pick three numbers that sum to 15

A: 8 4 5

B: 2 3 ?

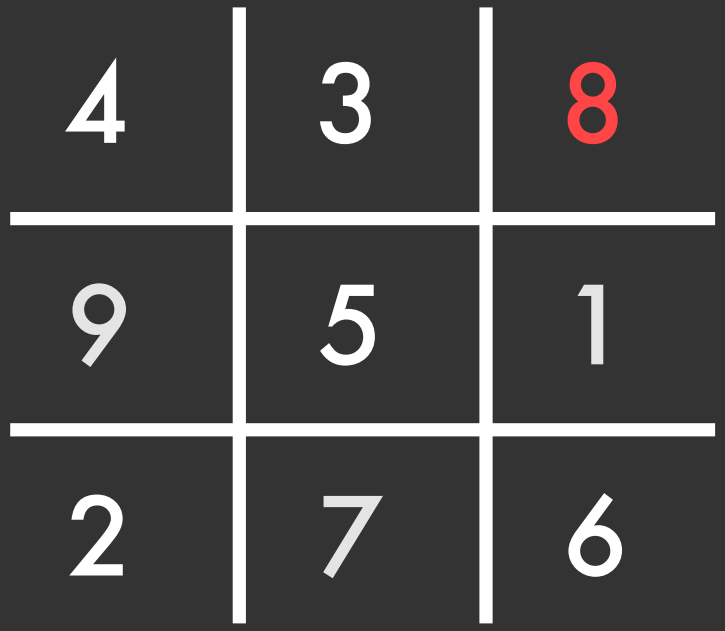
Problem Isomorphs

Problem Isomorph

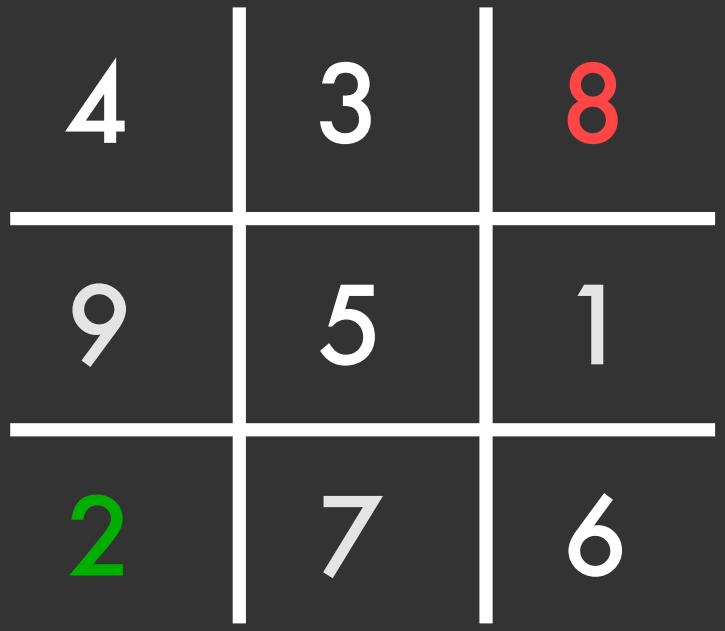
4	3	8
9	5	1
2	7	6

Magic Square: All rows, columns, diagonals sum to 15

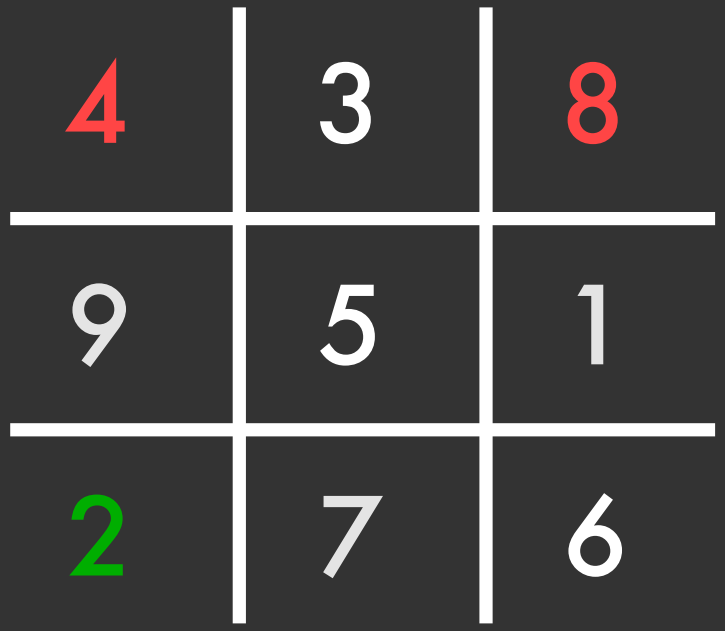
Transform to Tic-Tac-Toe



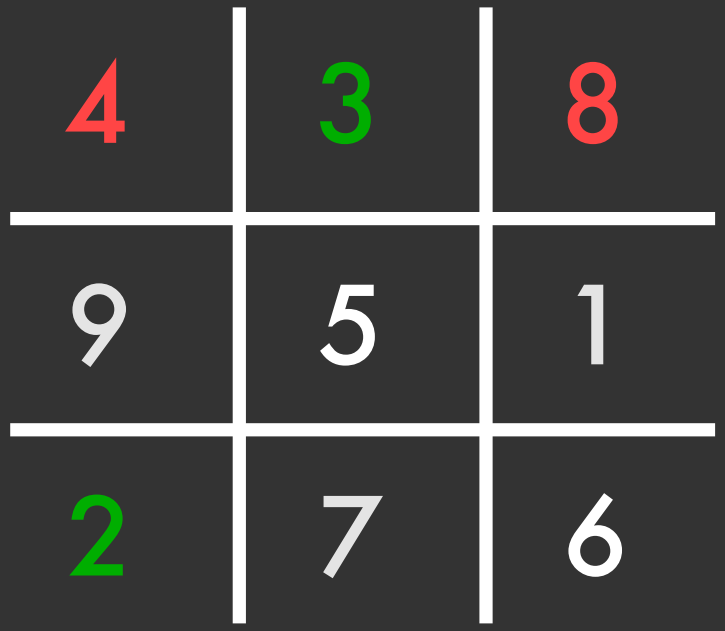
Transform to Tic-Tac-Toe



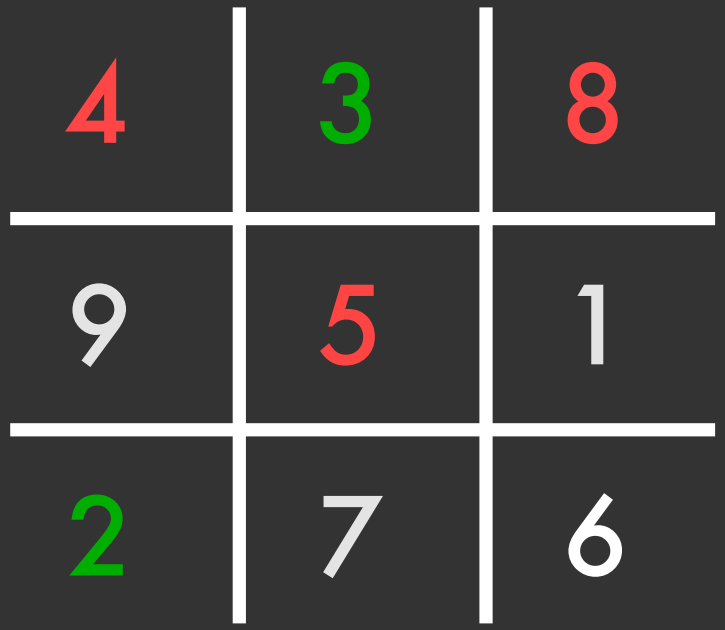
Transform to Tic-Tac-Toe



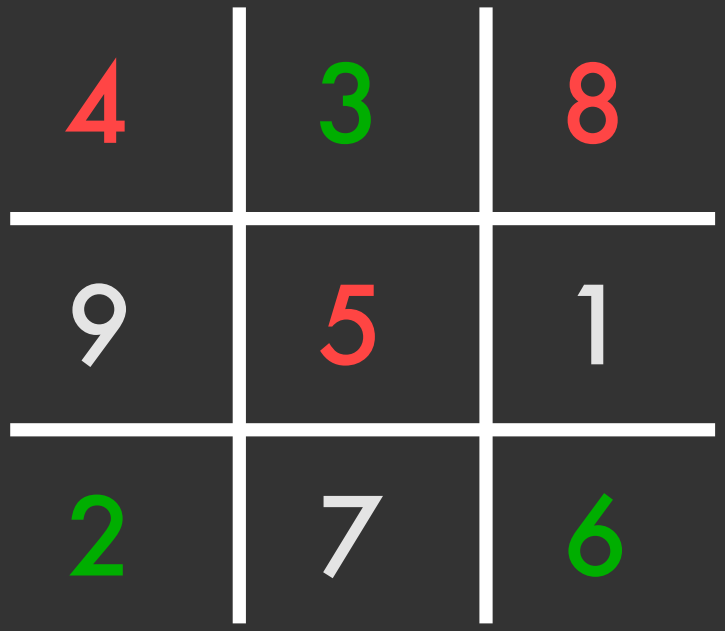
Transform to Tic-Tac-Toe



Transform to Tic-Tac-Toe



Transform to Tic-Tac-Toe



“Why is a Picture (Sometimes) Worth 10,000 Words”

Larkin and Simon,
Cognitive Science, 1987

Why?

Reduce memory load

- Working memory is limited
- Store information in the diagram

Reduce search time

- Pre-attentive (constant-time) search process
- Spatially-indexed patterns store the “facts”

Allow perceptual inference

- Map inference to pattern finding

The Representation Effect

Although two representations

may be equivalent,

one is often much “better”

for a given problem

“Better” means

- Faster
- Fewer errors
- Better comprehension
- …

“Number Representations”

Zhang and Norman

Number Representations

Counting – Tallying

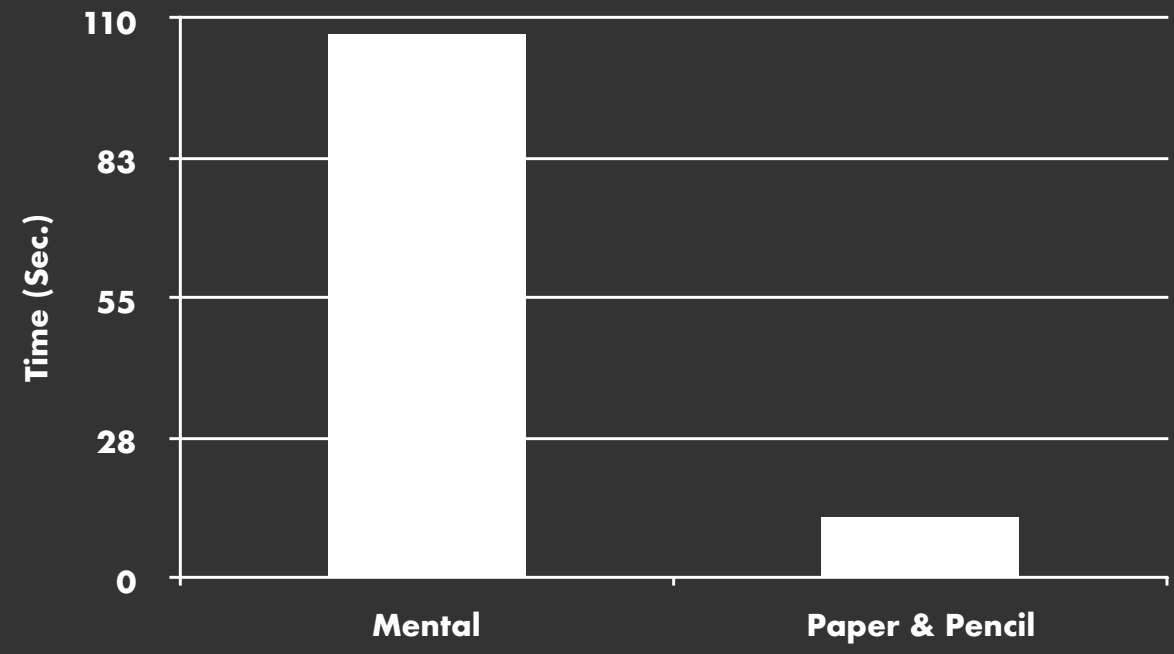
Adding – Roman numerals

XXIII + XII = XXXIIIII = XXXV

Multiplication – Arabic number systems

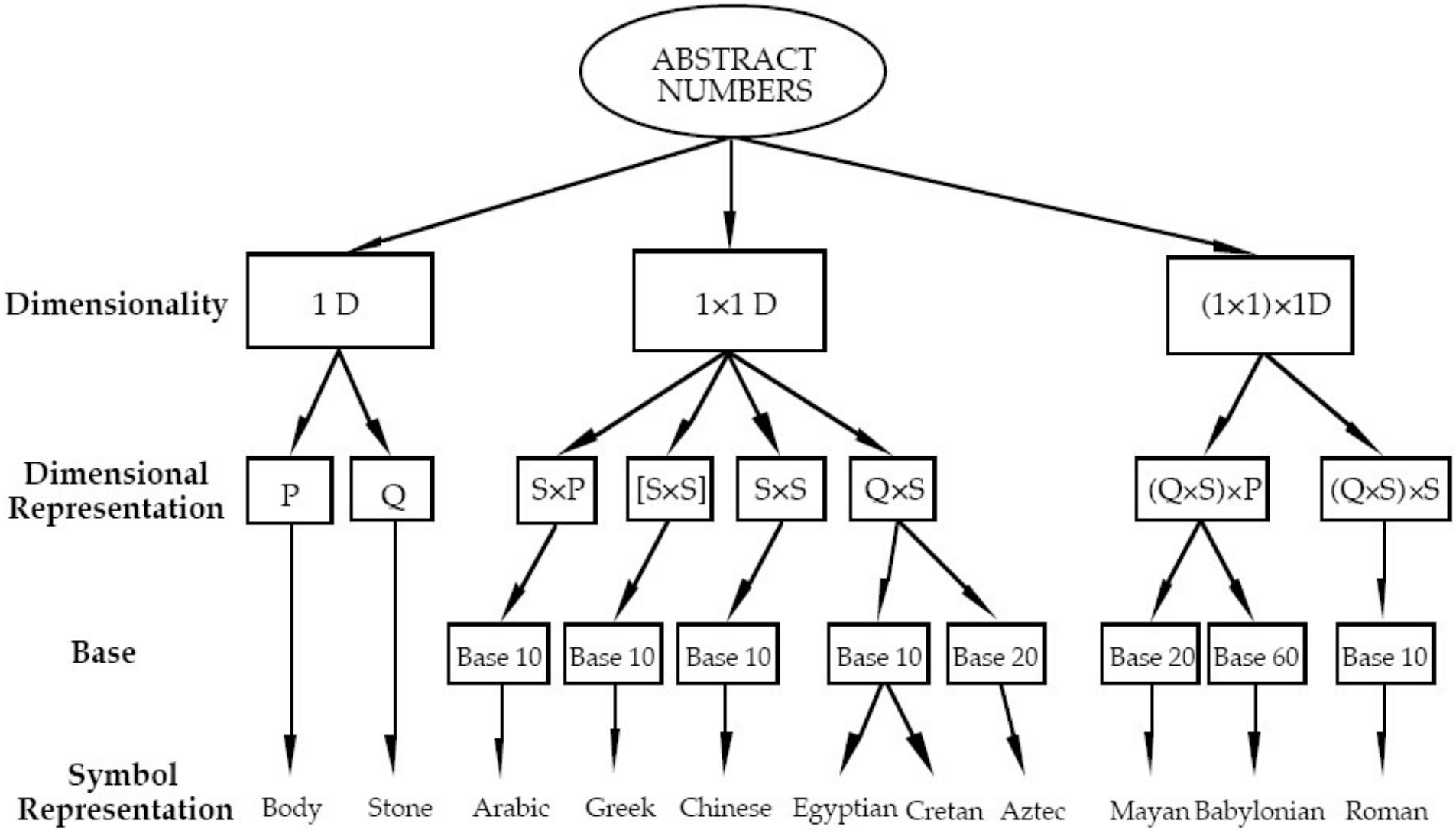
Long-Hand Multiplication

$$\begin{array}{r} 34 \\ \times\ 72 \\ \hline 68 \\ 238 \\ \hline 2448 \end{array}$$



From "Introduction to Information Visualization,"

Card, Schneiderman, Mackinlay



Zhang and Norman, The Representations of Numbers,

Cognition, 57, 271-295, 1996

Distributed Cognition

External (E) vs. Internal (I) process

		Roman	Arabic
1.	**Separate power & base**	**I**	**E**
2.	**Get base value**	**E**	**I**
3.	**Multiply base values**	**I**	**I**
4.	**Get power values**	**I**	**E**
5.	**Add power values**	**I**	**E**
6.	**Combine base & power**	**I**	**E**
7.	**Add results**	**I**	**E**

Arabic more efficient than Roman

Summary of Financial Performance

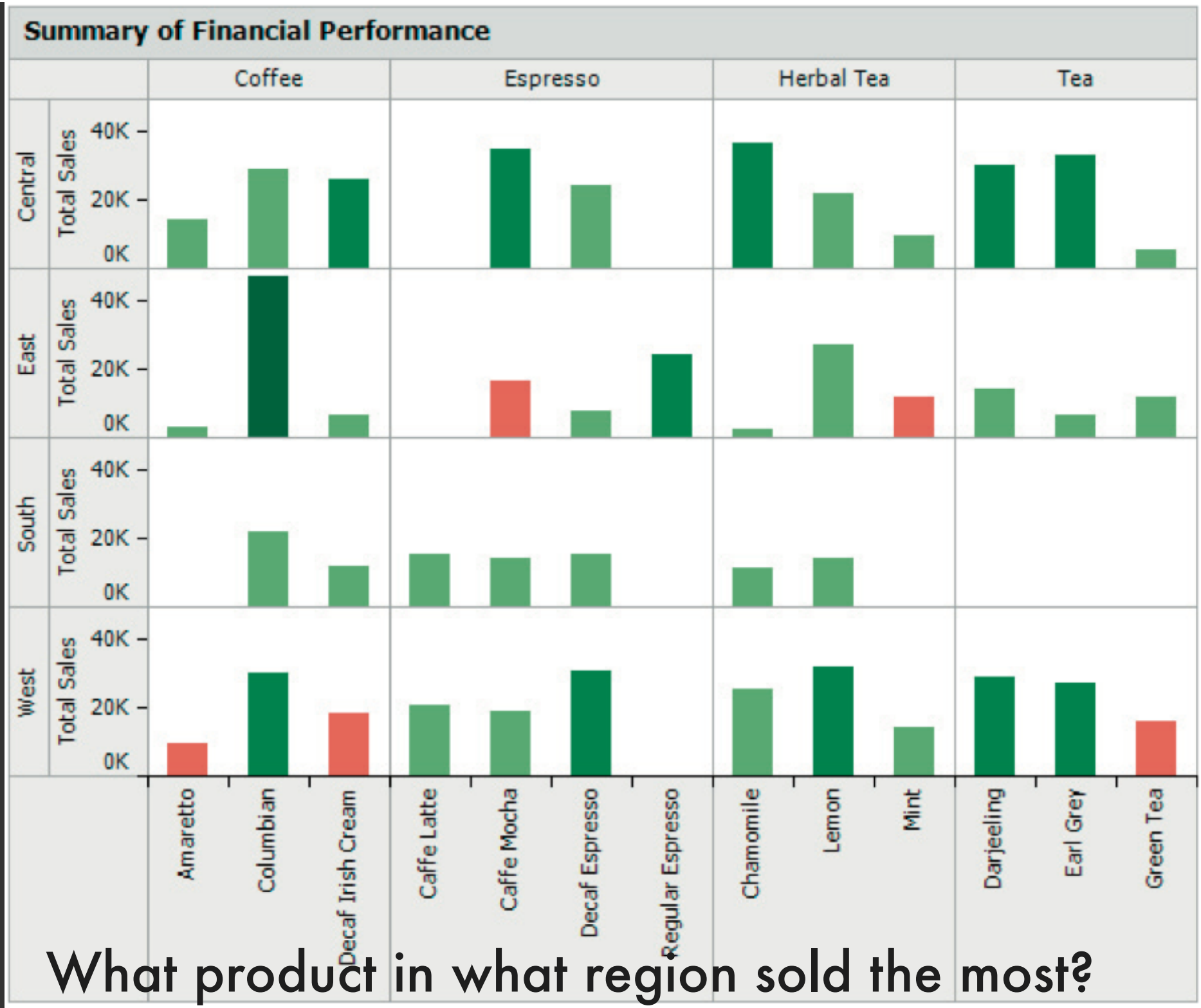
		Central		East		South		West	
		Total Sales	Total Profit	Total Sales	Total Profit	Total Sales	Total Profit	Total Sales	Total Profit
Coffee	Amaretto	$14,011	5,105	$2,993	1,009			$9,265	-1,225
	Columbian	$28,913	8,528	$47,386	27,253	$21,664	8,767	$30,357	11,253
	Decaf Irish Cream	$26,155	9,632	$6,261	2,727	$11,592	2,933	$18,235	-1,305
Espresso	Caffe Latte					$15,442	3,872	$20,458	7,502
	Caffe Mocha	$35,218	14,640	$16,646	-6,230	$14,163	5,201	$18,876	4,064
	DecafEspresso	$24,485	8,860	$7,722	2,410	$15,384	5,930	$30,578	12,302
	RegularEspresso			$24,036	10,062				
Herbal Tea	Chamomile	$36,570	14,434	$2,194	765	$11,186	3,180	$25,632	8,852
	Lemon	$21,978	6,251	$27,176	7,901	$14,497	2,593	$32,274	13,120
	Mint	$9,337	4,069	$11,992	-2,242			$14,380	4,330
Tea	Darjeeling	$30,289	10,772	$14,096	6,497			$28,769	11,780
	Earl Grey	$32,881	10,331	$6,505	3,405			$27,387	10,425
	Green Tea	$5,211	1,227	$11,571	5,654			$16,063	-7,109

How much mint tea was sold in the west?

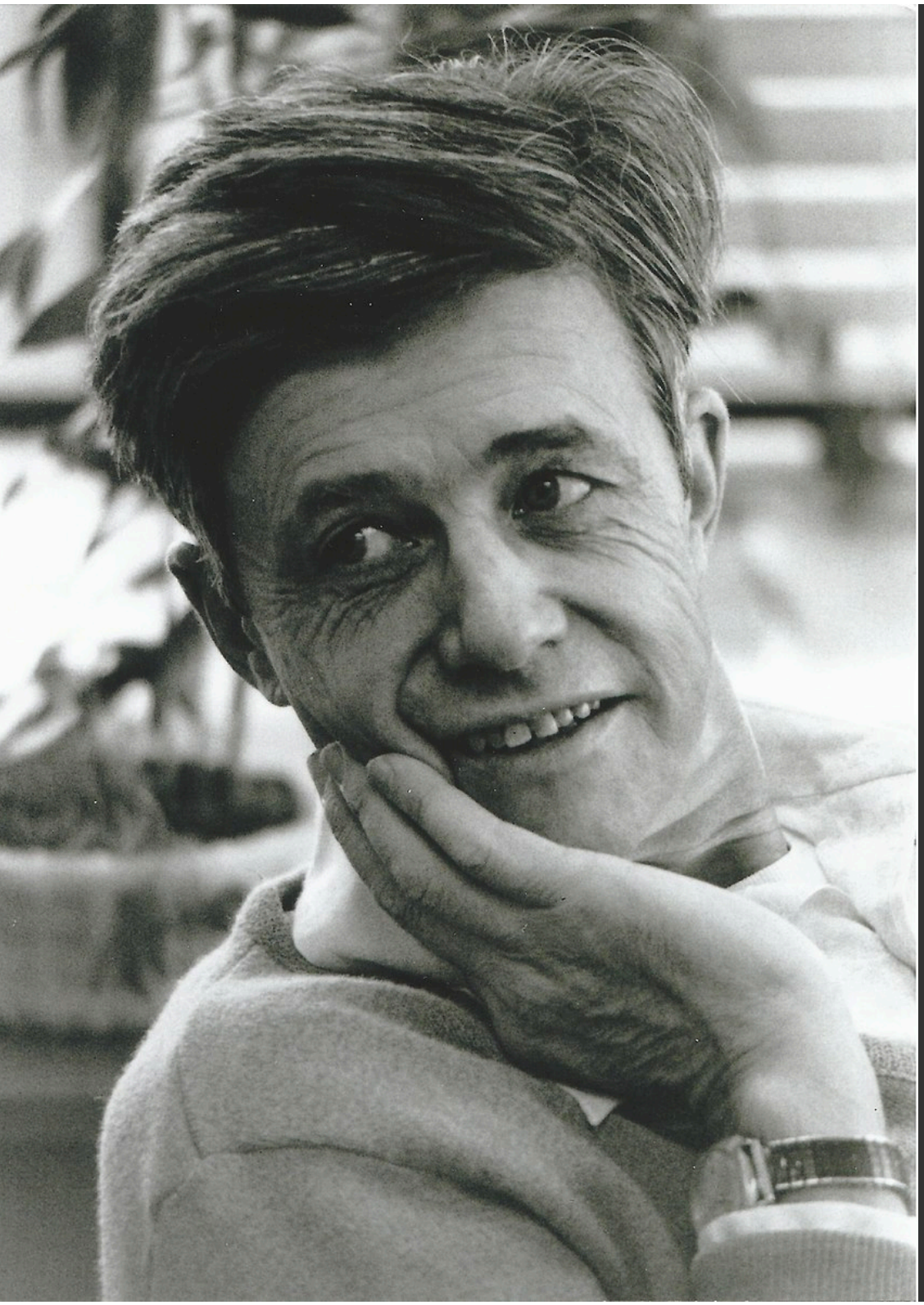
Summary of Financial Performance

		Central		East		South		West	
		Total Sales	Total Profit	Total Sales	Total Profit	Total Sales	Total Profit	Total Sales	Total Profit
Coffee	Amaretto	$14,011	5,105	$2,993	1,009			$9,265	-1,225
	Columbian	$28,913	8,528	$47,386	27,253	$21,664	8,767	$30,357	11,253
	Decaf Irish Cream	$26,155	9,632	$6,261	2,727	$11,592	2,933	$18,235	-1,305
Espresso	Caffe Latte					$15,442	3,872	$20,458	7,502
	Caffe Mocha	$35,218	14,640	$16,646	-6,230	$14,163	5,201	$18,876	4,064
	DecafEspresso	$24,485	8,860	$7,722	2,410	$15,384	5,930	$30,578	12,302
	RegularEspresso			$24,036	10,062				
Herbal Tea	Chamomile	$36,570	14,434	$2,194	765	$11,186	3,180	$25,632	8,852
	Lemon	$21,978	6,251	$27,176	7,901	$14,497	2,593	$32,274	13,120
	Mint	$9,337	4,069	$11,992	-2,242			$14,380	4,330
Tea	Darjeeling	$30,289	10,772	$14,096	6,497			$28,769	11,780
	Earl Grey	$32,881	10,331	$6,505	3,405			$27,387	10,425
	Green Tea	$5,211	1,227	$11,571	5,654			$16,063	-7,109

What product in what region sold the most?



What product in what region sold the most?



Notation as a Tool of Thought

Kenneth Iverson

Notation as a Tool for Thought

“The thesis of the present paper is that the advantages of executabiity and universality found in programming languages can be effectively combined, in a single coherent language, with the advantages offered by mathematical notation”

K. Iverson

Arithmetic and Algebra in APL (k)

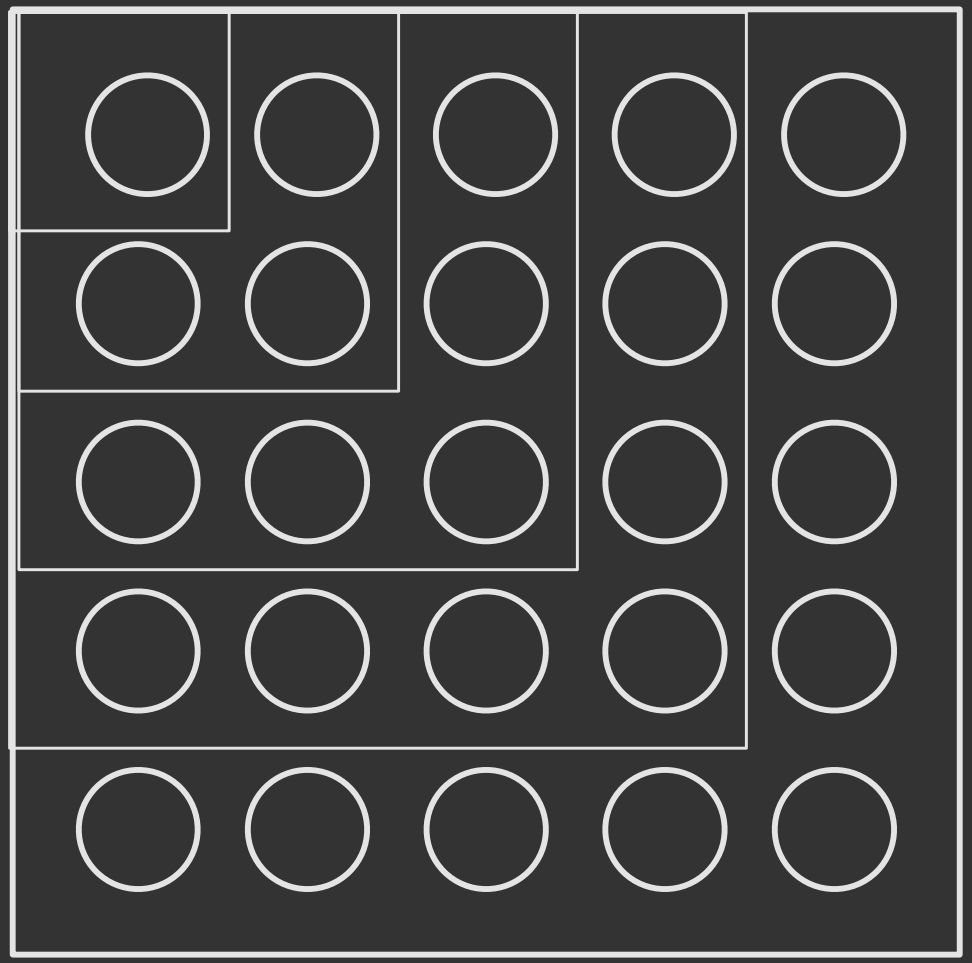
```
> k = 5
> til k
   0 1 2 3 4
> 1 + 2 * til k
   1 3 5 7 9
> +/ 1+2*til k          // 1 + 3 + 5 + 7 + 9
   25
> k*k
   25
```

Program Transformations as Proofs

```
+/ (1 + 2 * til k)          // odd numbers
+/ (1 + (til k) + (til k)) // def of multiplication
+/ (1 + (til k) + (reverse til k))
   // addition commutation
+/ (1 + k # (k-1))  // 0 1 2 + 2 1 0 = 2 2 2
+/ k#k              // k = k-1+1
k*k                 // k*k = +/ k#k
                          // 3*3 = +/ 3 3 3
```

K. Iverson, *Arithmetic*, 1991

Visual Proofs



Algebra

$1+3+5+7+9=5^2$

Characteristics of Notation

Executable mathematics

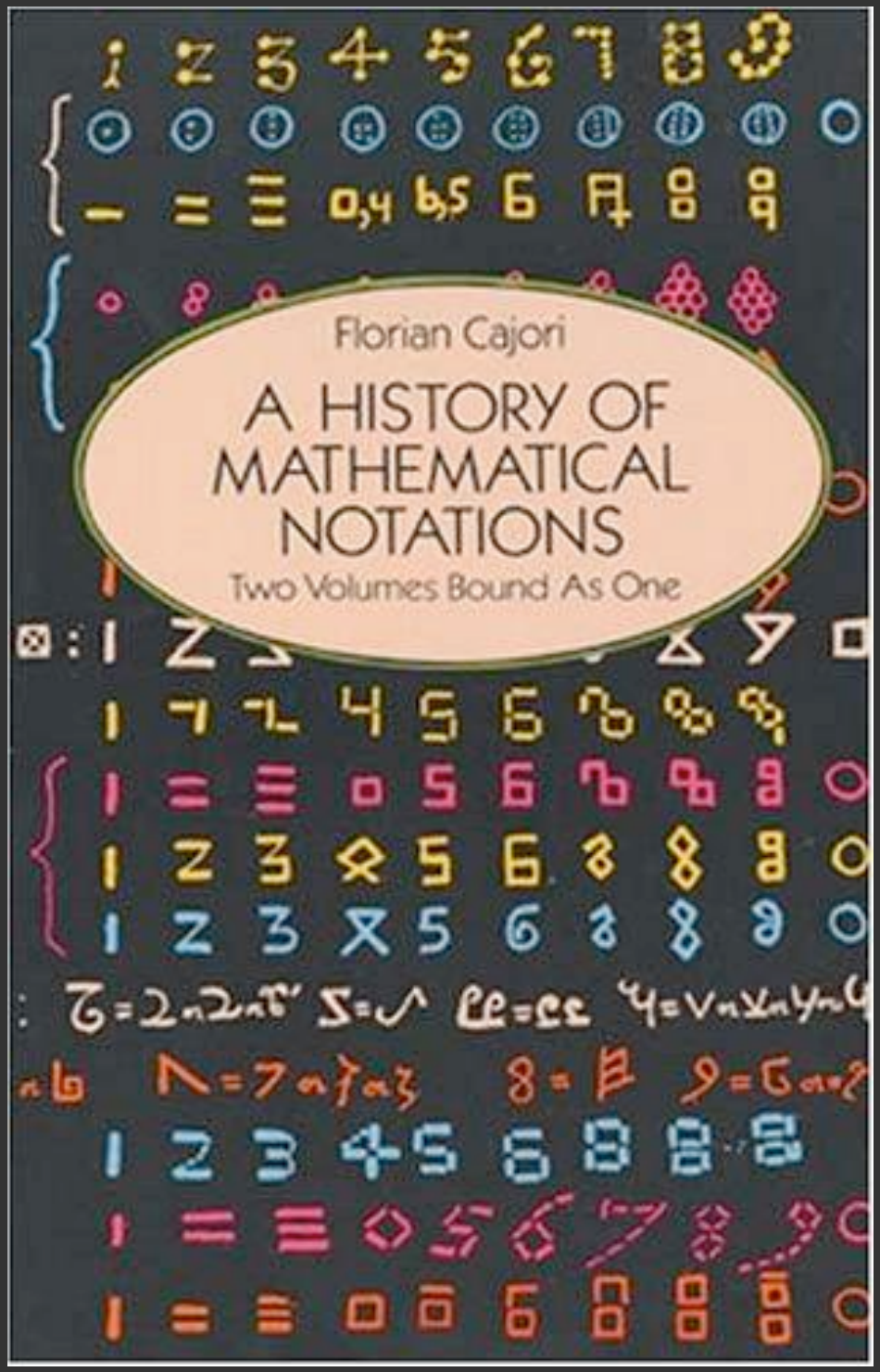
1 Ease of expression

2 Suggestivity (naturalness)

3 Subordinate detail (abstract)

4 Economy (concise)

5 Formal (proofs)



Florian Cajori (1859-1930)

History of Mathematics

University of California,

Berkeley

Mathematica

Rich character and symbol set (2500)

Multiple versions of a symbol to resolve ambiquity

- i
- double struck i = sqrt(-1)

Two-dimensional (over, subscript, ...)

StandardForm, TraditionalForm, ...

- 100 heuristic to go from traditional to standard

Input macros (map ascii to symbol, ->)

Supporting Notation

```
// Consistent Minimal syntax

// Lisp
(cond
  ((= n 10) (= m 1))
  ((> n 10) (= m 2) (= n (* n m)))
  ((< n 10) (= n 0)))

// S-expressions

// Special forms
```

```
// Smalltalk, Ruby
employee name first

// unary operators / methods
// parsed left to right
// left associative
//   = employee.name.first

// binary operators
// left associative, no precedence
1 + 2 * 3 = (1 + 2) * 3

// unary operators have precedence over
binary
1 + 4 sqrt = 1 + (4 sqrt)
```

```
// APL (k)
+/ 1 + 2 * til k

// monadic and dyadic functions
// operators (higher-order functions)
// right associative
// functions have same precedence
// operators > functions
+/ (1 + (2 * (til k)))
```

```
// Haskell

// function application with space
// sections
(+) 1 2

// left associative
// currying
(((+) 1) 2)

// precedence and associativity
infixl 6 +
```

```
// Haskell

// function application with space
// sections
(+) 1 2

// left associative
// currying
(((+) 1) 2)

// precedence and associativity
infixl 6 +

// unicode symbols
```

- From the `Prelude.Unicode` module
  - Values
    - `not` = `(¬)`
    - `(&&)` = `(∧)`
    - `(||)` = `(∨)`
    - `(==)` = `(≡)`
    - `(/=)` = `(≠)` = `(≢)`
    - `<=` = `≤` = `≯`
    - `>=` = `≥` = `≮`
    - `pi` = `π`
    - `(/)` = `(÷)`
    - `(*)` = `(·)`
    - `(.)` = `(∘)`
    - `(++)` = `(⧺)`
    - `elem` = `(∈)`
    - `notElem` = `(∉)`
    - `undefined` = `(⊥)`
  - Types
    - `Integer` = `ℤ`
    - `Rational` = `ℚ`

The Incredible Convenience of Mathematica Image Processing

Theodore Gray

Cajori

Typographic *purity*

- ***See Typographics Style***

Unicode character classes

- https://en.wikipedia.org/wiki/Unicode_character_property

Common lexical tokens

Base operator-precedence (Pratt) parser

User-defined recursive descent parser

List IR