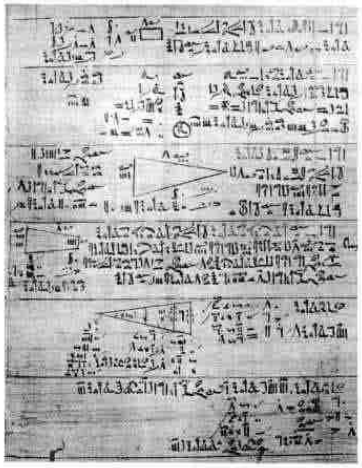
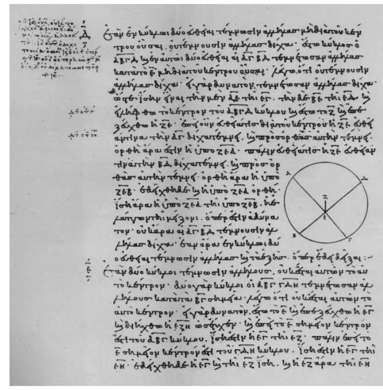
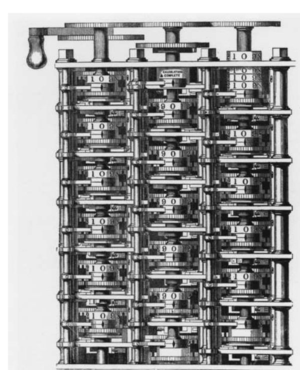




Analysis of Algorithms —

Between Mathematics and Computer Science

Philippe Flajolet, INRIA Rocquencourt, F.

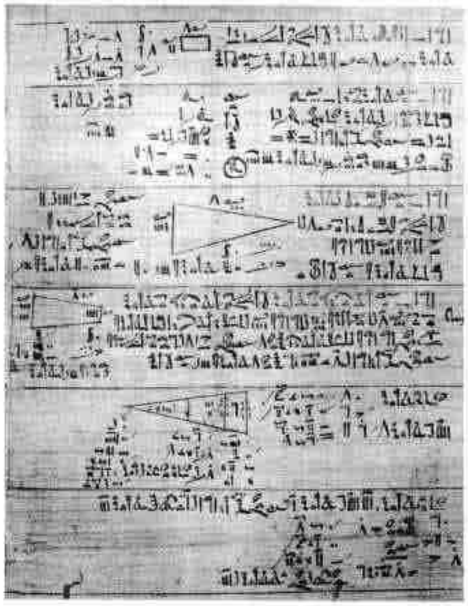

FIRST,

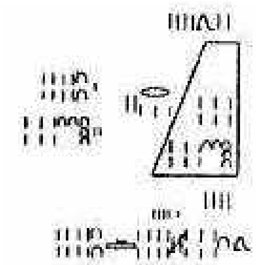
A glimpse of history . . .

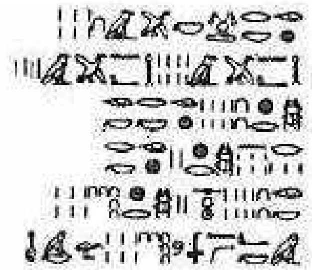
Mathematics and Computing, i.e., *algorithms*,
= a joint enterprise since the dawn of history.

Thesis: (i) Conceptual advances lead to more complex and efficient algorithms.
(ii) Computer age obeys similar principle?

Rhind papyrus (ca 1650BC $\rightsquigarrow$ 1900BC)







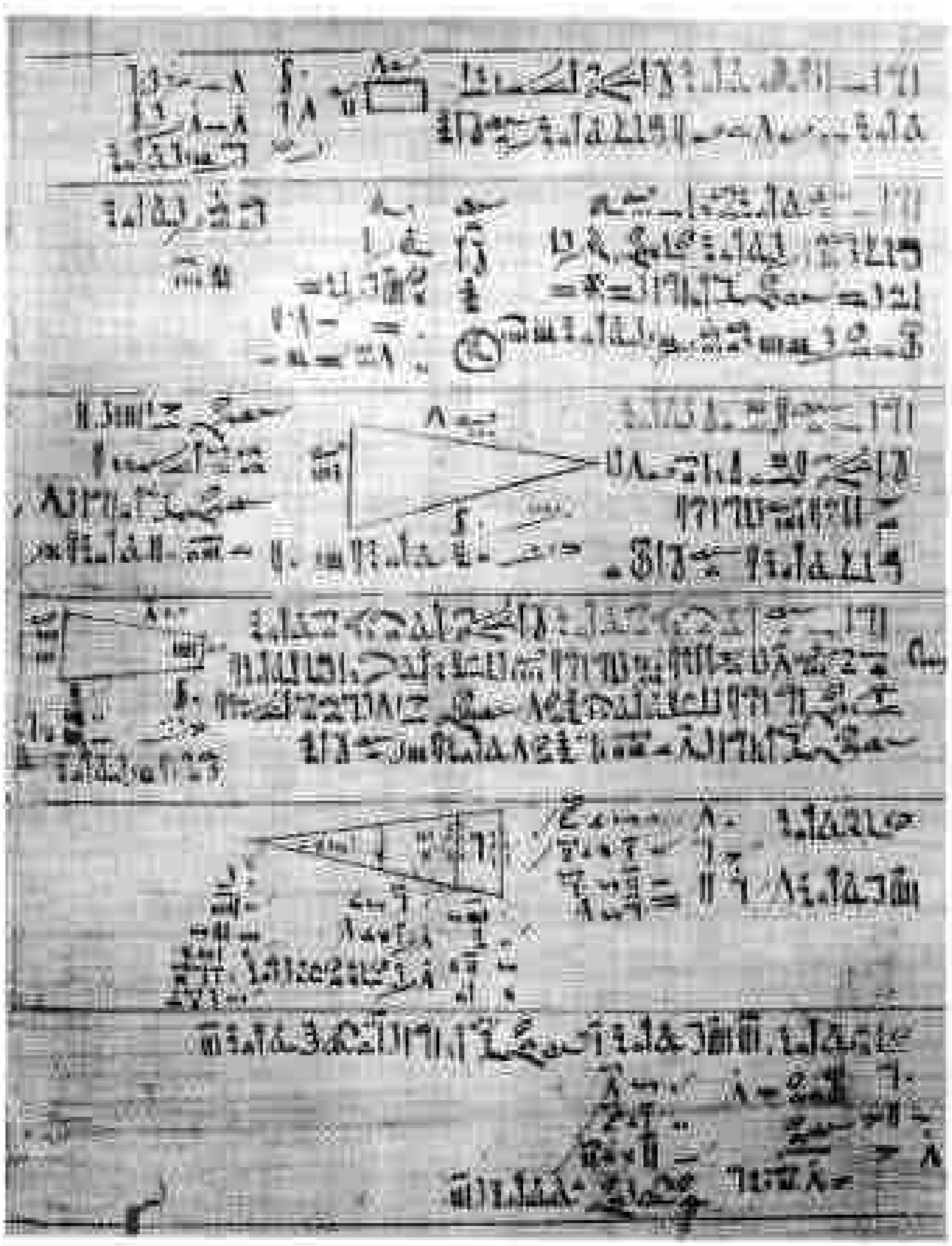
Egyptians *knew* binary representations and technique of "binary powering"!

$$41 \times x = \overbrace{\left(1 + 2^3 + 2^5\right)}^{1\,+\,8\,+\,32} \times x = 1x + 2^3x + 2^5x.$$

41	$\times$	59	
1		59	$\surd$
2		118	
4		236	
8		472	$\surd$
16		944	
32		1888	$\surd$
		= 2419	

RSA, PGP:

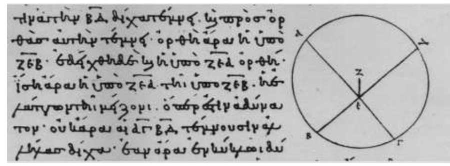
$$x^{41} = \boxed{x^1} \cdot [x^2 \cdot x^4] \cdot \boxed{x^8} \cdot [x^{16}] \cdot \boxed{x^{32}}.$$



The Rhind papyrus contains eighty-seven problems. The papyrus, a scroll about 6 metres long and 1/3 of a metre wide, was written around 1650 BC by the scribe Ahmes who states that he is copying a document which is 200 years older.

Computing *without* Computers!

Geometry



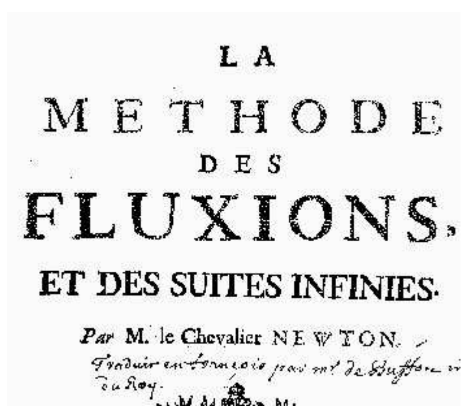
• Euclid (325BC–265BC) discovers *Euclid's algorithm* and formalizes *geometry*. Archimedes (287BC–212BC) discovers that *π is computable*; cf Viète (1540–1603):

$$\frac{\pi}{2} = \frac{2}{\sqrt{2}} \frac{2}{\sqrt{2+\sqrt{2}}} \frac{2}{\sqrt{2+\sqrt{2+\sqrt{2}}}} \cdots$$

Arithmetics & Algorithms

• Al Kwarizmi (780–850) gives complete set of rules, an *"algorithm"* for the four operations on "hindi" numerals.

Calculus


LA
METHODE
DES
FLUXIONS,
ET DES SUITES INFINIES.
Par M. le Chevalier NEWTON.

• Newton (1643–1727) *"De Methodis Serierum et Fluxionum"* = Newton's algorithm; "computer algebra"

Let $P(x, y) = 0$; determine $\frac{d}{dx} y(x)$? Cf: Newton 1671; here, Buffon's translation.

METHODE

PROBLEME I.

Etant donnée la Relation des Quantités Fluentes, trouver la Relation de leurs Fluxions.

SOLUTION.

I. DISPOSEZ l'Equation par laquelle la Relation donnée eſt exprimée ſuivant les Dimenſions de l'une de ſes Quantités Flueutes x par exemple, & multipliez ſes Termes par une Progreſſion Arithmetique quelconque, & enſuite par $\frac{\dot{x}}{x}$ faites cette Opération ſéparément pour chacune des Quantités Fluentes ; après quoi égalez à zero la ſomme de tous les produits, & vous aurez l'Equation cherchée.

II. EXEMPLE I. Si la Relation des Quantités Fluentes x & y eſt $x^3 - ax^2 + axy - y^3 = 0$, diſpoſez d'abord les Termes ſuivant x, & enſuite ſuivant y, & multipliez-les comme vous voyez.

Multipliez	x^3	$-ax^2$	$+axy$	$-y^3$	$-y^3$	$+axy$	$-ax^2 + x^3$	
par	$\frac{3\dot{x}}{x}$.	$\frac{2\dot{x}}{x}$.	$\frac{\dot{x}}{x}$.	o .	$\frac{3\dot{y}}{y}$.	$\frac{\dot{y}}{y}$.	o	
Vous aurez	$3\dot{x}x^2$	$-2a\dot{x}x$	$+a\dot{x}y$	*	$-3\dot{y}y^2$	$+a\dot{y}x$	*	

la ſomme des produits eſt $3\dot{x}x^2 - 2a\dot{x}x + a\dot{x}y - 3\dot{y}y^2 + a\dot{y}x$, qui étant égalée à zero, donne la Relation des Fluxions $\dot{x}$ & $\dot{y}$; car ſi vous donnez à volonté une valeur à x, l'Equation $x^3 - ax^2 + axy - y^3 = 0$, donnera la valeur de y ; ce qui étant déterminé, l'on aura $\dot{x} : \dot{y} :: 3y^2 - ax : 3x^2 - 2ax + ay$.

- Euler, Gauß and others apeal to *computation* a lot!

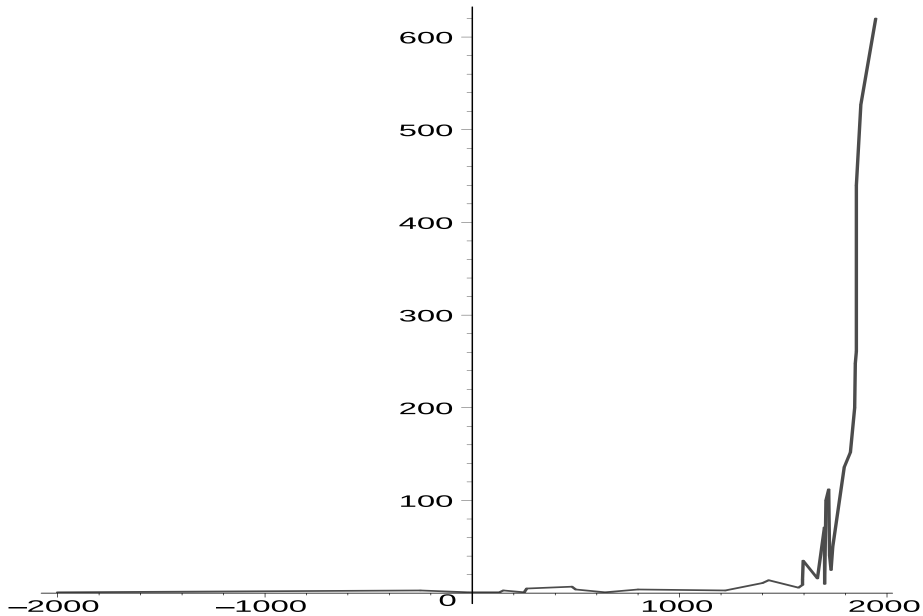
— Mathematics and computing largely progress together till the XIX–th century.

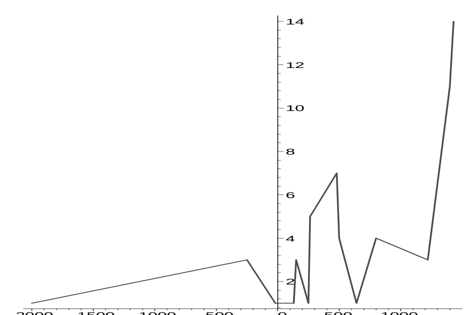
Computing *without* Computers!—π

1 Rhind papyrus	2000 BC	3.16045 (= 4(8/9)2)
2 Archimedes	250 BC	3.1418 (average of the bounds)
3 Vitruvius	20 BC	3.125 (= 25/8)
4 Chang Hong	130	3.1622 (= sqrt10)
5 Ptolemy	150	3.14166
6 Wang Fan	250	3.155555 (=142/45)
7 Liu Hui	263	3.14159
8 Tsu Ch'ung Chi	480	3.141592920 (= 355/113)
9 Aryabhata	499	3.1416 (=62832/2000)
10 Brahmagupta	640	3.1622 (= sqrt10)
11 Al-Khwarizmi	800	3.1416
12 Fibonacci	1220	3.141818
13 Madhava	1400	3.14159265359
14 Al-Kashi	1430	3.14159265358979
15 Otho	1573	3.1415929
16 Viète	1593	3.1415926536
17 Romanus	1593	3.141592653589793
19 Van Ceulen	1596	35 **D**
20 Newton	1665	16 **D**
21 Sharp	1699	71 **D**
22 Seki Kowa	1700	10 **D**
24 Machin	1706	100 **D**
25 De Lagny	1719	127 **D**, 112 correct
26 Takebe	1723	41 **D**
27 Matsunaga	1739	50 **D**
28 von Vega	1794	140 **D**, 136 correct
29 Rutherford	1824	208 **D**, 152 correct
30 Strassnitzky, Dase	1844	200 **D**
31 Clausen	1847	248 **D**
32 Lehmann	1853	261 **D**
33 Rutherford	1853	440 **D**
34 Shanks	1874	707 **D**, 527 correct

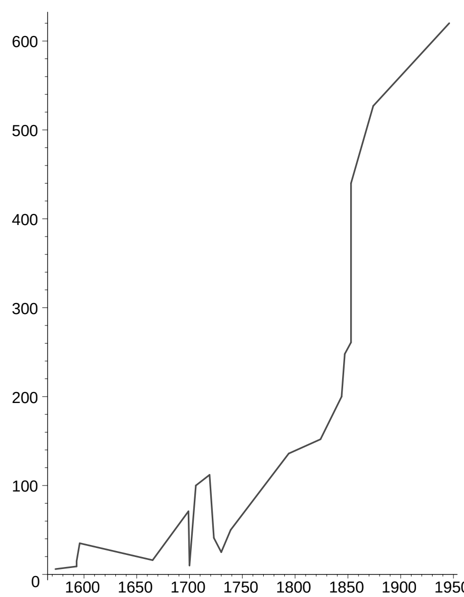
Source: `http://www-gap.dcs.st-and.ac.uk/~history/HistTopics/Pi_chronology.html`

π: From -2000 to 1946 (# of **D**igits)





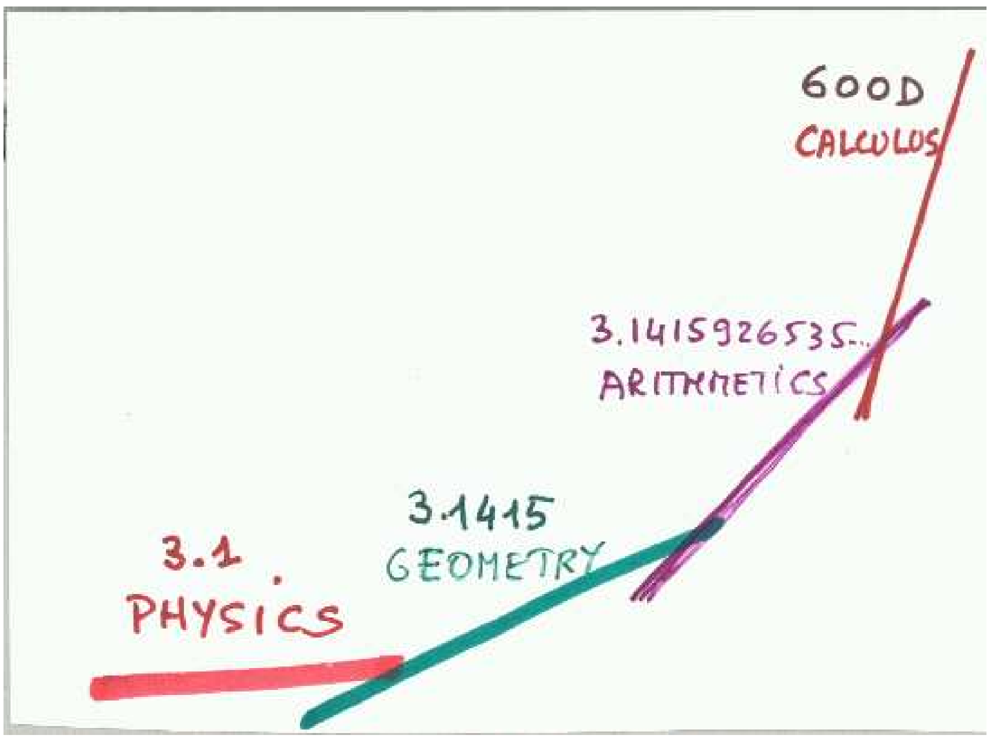
1–14 **D**, till 1500



15–620 **D**, after 1500

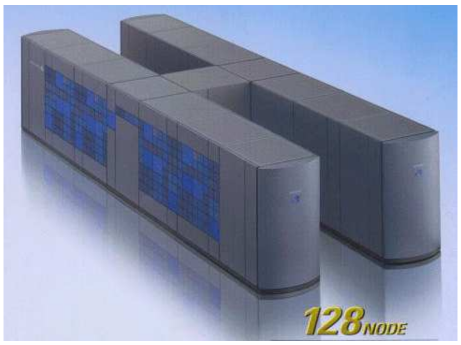
PROGRESS = *Geometry + Arithmetics + Analysis*.

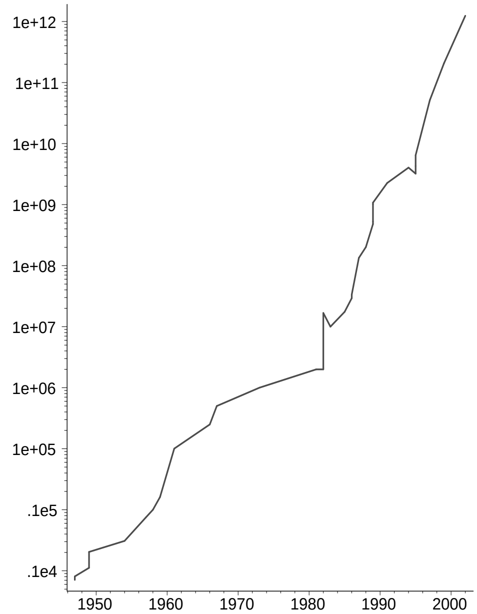
PROGRESS = *Geometry + Arithmetics + Analysis*.



Computing *with* Computers! — π

ENIAC, 1949: 1120**D**; 1000 IPS; Supercomputer 2002: $2 \cdot 10^{12}\mathbf{D}$; 10^{12} IPS (Instruction Per Second)





Moore's law

ENIAC 1949 : $\dfrac{1120\mathbf{D}}{1000 IPS} = 1.12;$ Kanada 2002 : $\dfrac{2 \cdot 10^{12}\mathbf{D}}{10^{12} IPS} = 2.00$

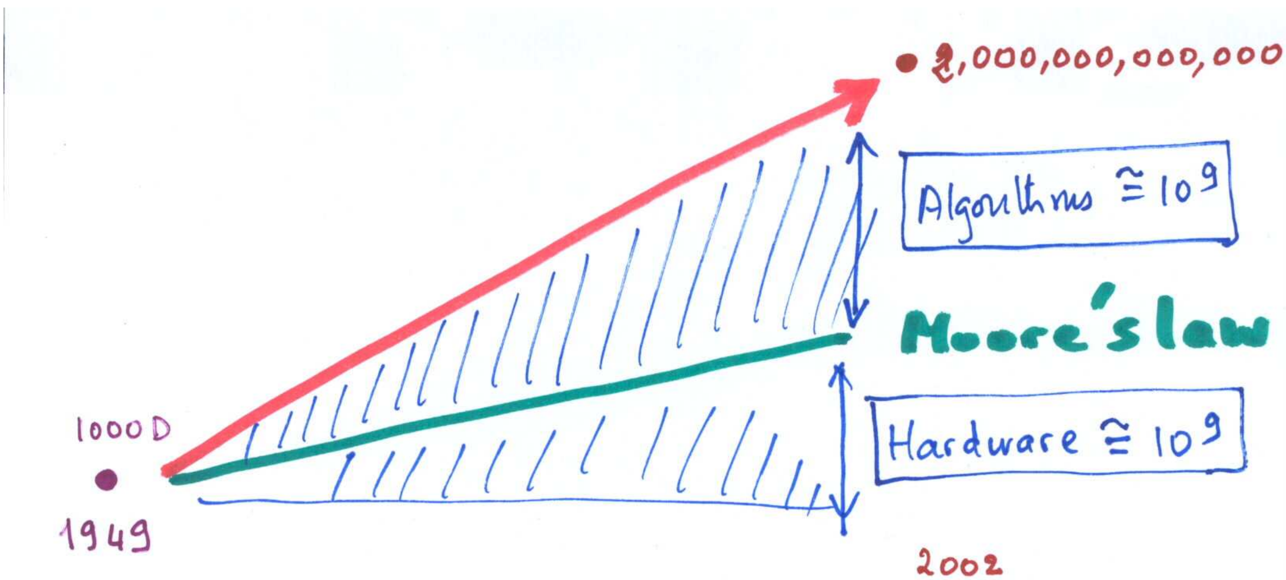
$\cdots$ *is only* **half** *of the story.*

Computation cost is **superlinear** $\Longrightarrow$ **better algorithms** are **needed**!!

Initially: $\approx \mathcal{O}(n^2)$, $\dfrac{\pi}{4} = 4 \arctan \dfrac{1}{5} - \arctan \dfrac{1}{239}.$

- Subquadratic multiplication (Karatsuba)
- Fast Fourier transform
- Arithmetic-geometric mean; elliptic functions . . .
- Superquadratically convergent algorithms

Finally: $\approx \mathcal{O}\left(n(\log n)^2\right)$



1,000,000,000,000
Algorithms ≅ 10^9
Moore's law
Hardware ≅ 10^9
1000 D
1949
2002

An aside: **‘ miraculous’ Bailey-Borwein-Plouffe alg.**

$$\pi = \sum_{n=0}^{\infty} \left(\frac{4}{8n+1} - \frac{2}{8n+4} - \frac{1}{8n+5} - \frac{1}{8n+6} \right) \left(\frac{1}{16} \right)^n .$$

The forty–trillionth bit of Pi is ‘0’
101 [0] 0000 1111 1001 1111 1111 0011 0111 0001
= A0F9FF371D17593E

$$\pi = \int_0^{1/\sqrt{2}} \frac{4\sqrt{2} - 8x^3 - 4\sqrt{2}x^4 - 8x^5}{1 - x^8} \, dx.$$

Experimental maths in the computer age: Found originally by PSLQ algorithm, finding dependencies between high precision evaluations applied to an inspired guess.
Cf. CECM site on *Experimental Mathematics* at Vancouver & Borwein’s pages.
A curiosity

$$B := 4 \sum_{k=1}^{500\,000} \frac{(-1)^{k-1}}{2k-1},$$

$$B = 3.14159065358979324046264338326950288419729139937$$

$$\pi = 3.14159265358979323846264338327950288419716939937$$

Yet another case

Integer factorization challenge

The problem of decomposing $15 = 3 \times 5$ is *not* known to be in class Polynomial time.

Triggered by Public Key Cryptosystems based on arithmetic strctures, a la RSA.

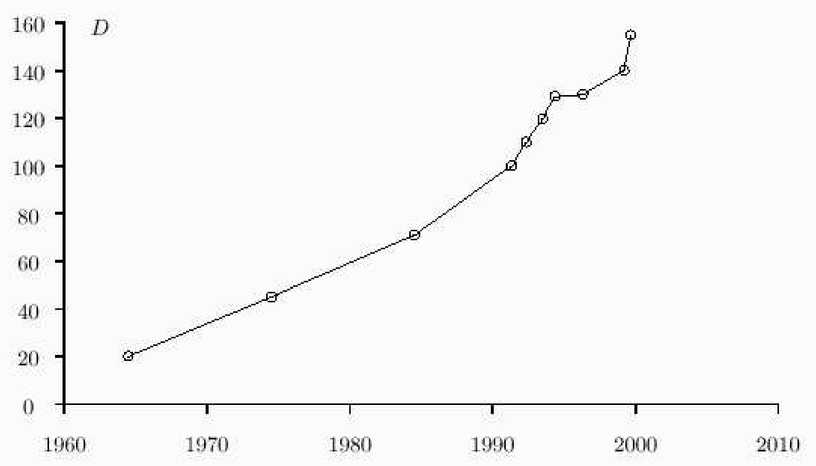


Figure 3: Size of "general" number factored versus year

(©Richard Brent.)

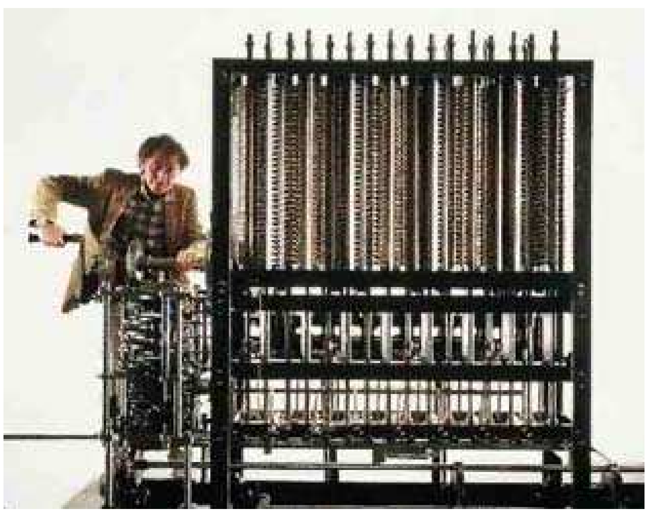
Probabilistic algorithms start largely with Rabin in 1976: here (almost) all the algorithms are *randomized* —they make bets...

Analysis of Algorithms = an indispensable companion!

Some algorithms are more efficient than others.
— By how much ? Why? $\leadsto$ Optimizations

"Subliminal" in classical math.
— Trial division for factoring and Erastothenes' sieve are costly.
— Newton's algorithm for root finding doubles the number of digits at each stage $\neq$ fixed-point iteration only adds a fixed amount.

— Charles Babbage (1837)



With respect to the time employed $\cdots$ in turns of the handle:

$$\frac{20(12+n)+15(p+4)}{382} \quad \text{for Mult without stepping}$$

$$\text{and} \quad \frac{20(12+n)+15(p+d+4)}{382} \quad \text{for Mult with stepping}$$

Burks, Goldstine, von Neumann, 1946 (US Army)

"The logical design of an electronic computing instrument"

$\cdots$ "We shall show that for a sum of binary words, each of length n, the length of the largest carry sequence is on the average not in excess of ${}^2\log n$."

$\leadsto$ Feller, Knuth: Runs of good luck in coin tossings...

Next . . .

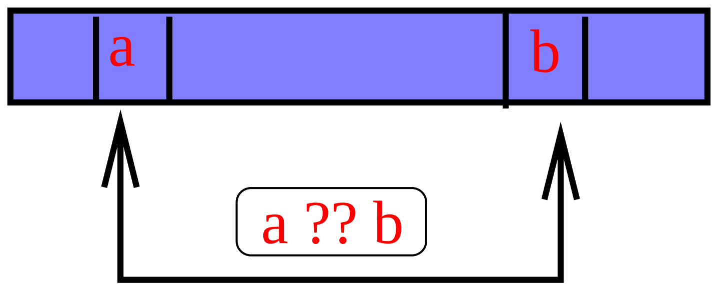
The Saga of Digital Trees

1. Pioneers

1950's: Scientific computing meets information processing $\rightsquigarrow$ *non-numerical data*, esp. Sorting & Searching.

First algorithms deal with sorting and searching.

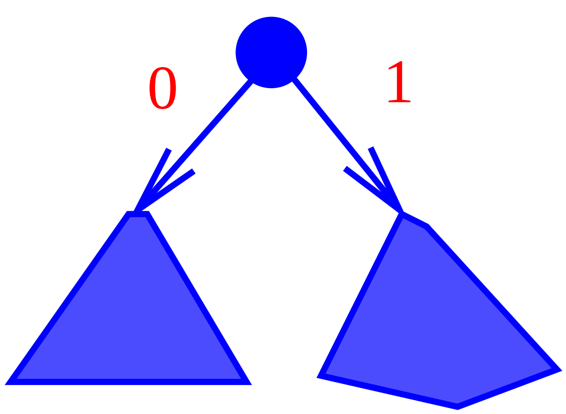
Radix-exchange sort (H&I)



Compare-exchange based on successive *bits* of data.

```
place 0's on left, 1's on right;
recurse.
```

The trie splitting process (Fredkin)



Separate recursively based on successive *bits* of data.

Journal of the ACM Vol. 6 (April 1959)

Radix Exchange—An Internal Sorting Method for Digital Computers*

PAUL HILDEBRANDT AND HAROLD ISBITZ

System Development Corporation, Santa Monica, California

This note describes a new technique—Radix Exchange. The technique is faster than Inserting by the ratio $(\log_2 n)/n$ Its speed compares favorably with internal merging and it has the significant advantage of requiring essentially no working area. . .

Communications of the ACM Vol. 3 (August 1960)

Techniques

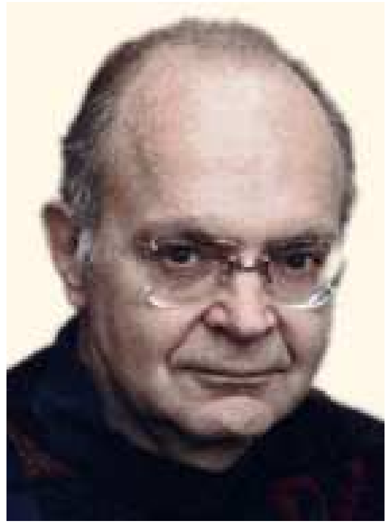
Trie Memory*

EDWARD FREDKIN, *Bolt Beranek and Newman, Inc., Cambridge, Mass.*

Nexuses and Nexus Chains

In order to permit more general description of trie memory, it is helpful to substitute, for the system of successive addresses used in connection with the illustrations in Fig. 1, a system of directed connections that we may call nexuses. A brief explication of nexuses and nexus chains will facilitate further discussion of trie memory.

Don Knuth (b. 1938)

???  ???

What is the number of turns of the handle?

At CalTech around 1965, cooperation of Knuth & de Bruijn
In *The Art of Computer Programming* 1973

Page 131 of Knuth's TAOCP, Vol. 3 (1973) — The original derivation

♣ Decompose $\Longrightarrow$ Divide & Conquer recurrence:

$$C_n = n + \sum_{k=0}^{n} \frac{1}{2^n} \binom{n}{k} (C_k + C_{n-k}) \, .$$

♢ Solve binomial recurrence & reorganize.

♡ Asymptotics: cleverly use Gamma function

$$e^{-x} = \frac{1}{2i\pi} \int_{1-i\infty}^{1+i\infty} \Gamma(s)\, x^{-s} ds$$

♠ *Miraculous factorizations occur, residues fly all around, and . . .*

The Big Theorem of P. 131± of Knuth's Vol. 3

• Tries and Radix-exchange sort have expected cost
(path length, bit comparisons)

$$\sim \boxed{n \log_2 n} + n \left(\frac{\gamma - 1}{\log 2} - \frac{1}{2} + f(n) \right)$$

"where $f(n)$ is the rather strange function $\cdots$
Furthermore

$$f(n) < 0.000001725$$

thus we may safely ignore $f(n)$ for practical purposes."

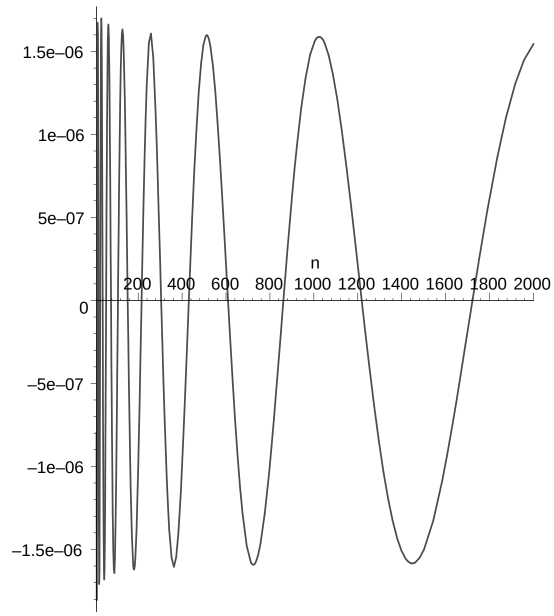
• Size has expectation (with fluctuations!)

$$\sim \boxed{\frac{n}{\log 2}} + n \cdot \widehat{f}(n)$$

$$f(n) = \frac{1}{\log 2} \sum_{k \neq 0} \Gamma \left(-1 - \frac{2ik\pi}{\log 2} \right) \exp \left(2ik\pi \log_2 n \right)$$

Criticisms

Fluctuations$\approx 10^{-6}$:



♠ A complicated math exercise. An isolated problem.

♠ An expected outcome ($\pm$): $O(n \log n)$ by *easy* probabilistic argument.

♠ A useless answer with 10^{-6} fluctuations!

♠ With Moore's law, anyhow, etc.

The Saga of Digital Trees

2. Analysis

Some "modern" views: Trabb Pardo 1978, Greene 1980, F.–Régnier–Sedgewick–Sotteau 1985, F.–Gourdon-Dumas 1995.

Methodological advances

Symbolic methods: Combinatorics is reflected by *algebra of generating functions*

Mainstream methods of enumerative combinatorics (≥ 1980) replace recurrences.

$$\left\{ f_n \right\} \longrightarrow f(z) := \sum_n f_n z^n.$$

$\leadsto$ Difference equations for expected trie costs:

$$\phi(z) \quad = \quad 2e^{z/2}\phi\left(\frac{z}{2}\right) + \text{toll}\,(z).$$

Semiclassical: Iteration, coefficient extraction, . . .

Methodological advances

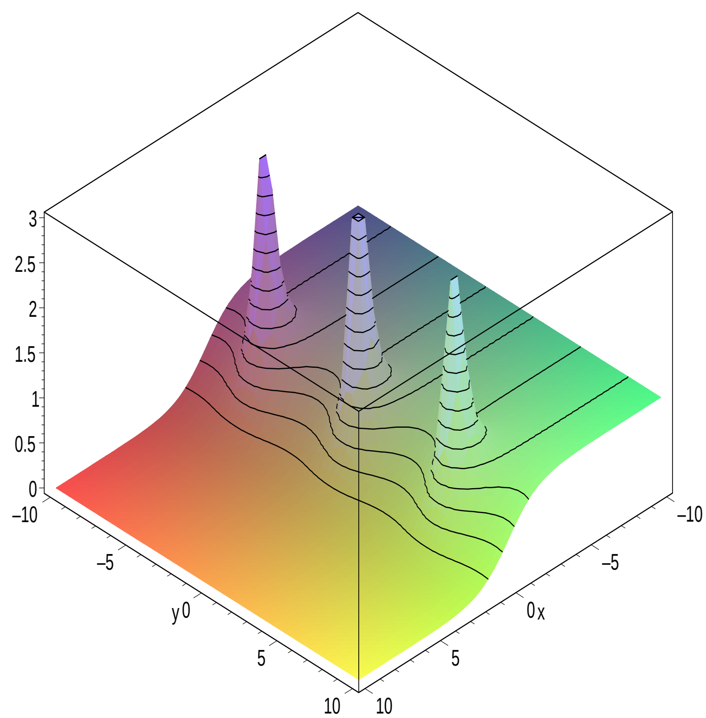
♡ Mellin transforms

$$f \xrightarrow{\mathcal{M}} f^\star := \int_0^\infty f(x) x^s \frac{dx}{x}$$

Real asymptotics from *complex* singularities.

Factorizes linear superposition of models

$$\sum \lambda_k f(\mu_k x) \quad \mapsto \left(\sum \lambda_k \mu_k^{-s} \right) \cdot f^\star(s).$$



$$\frac{\Gamma(s)}{1 - 2^{-s}}$$

Work from 1965++ yields a **systematic approach**

Algorithms

$\Downarrow$

Algebra of Costs Gen. Functions

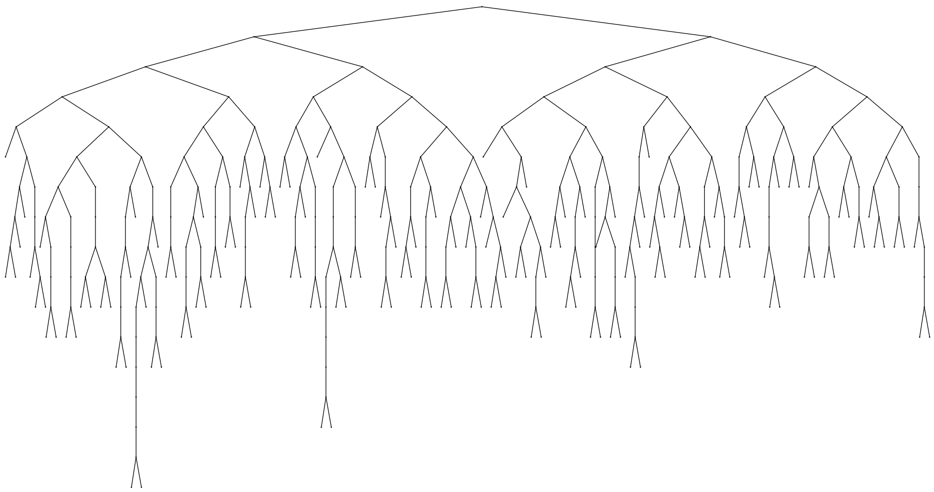
$\Downarrow$

Asymptotic estimates from singularities

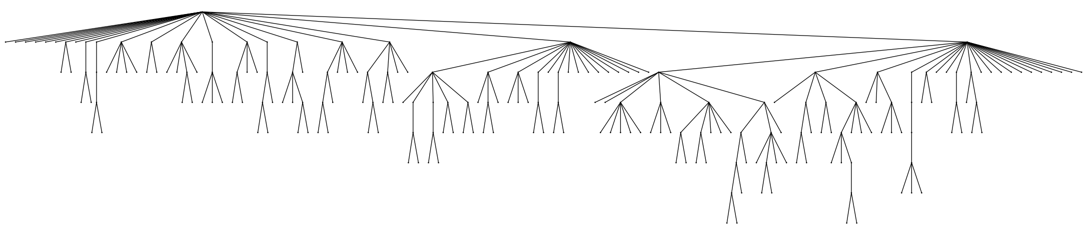
applicable to a major combinatorial process of computer science.

Knuth's and others' results inform us on *shape* of certain trees:

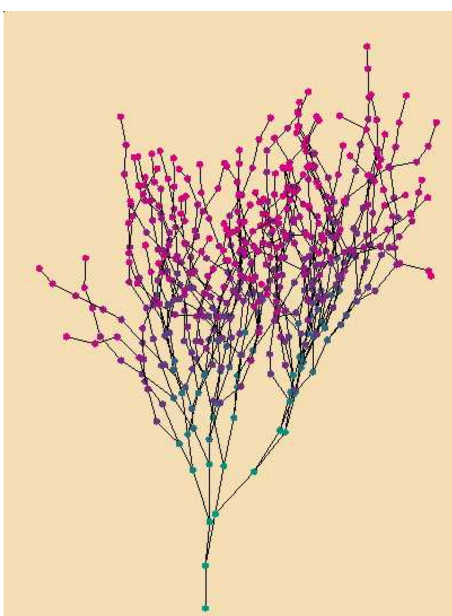
Binary trie (uniform bits)

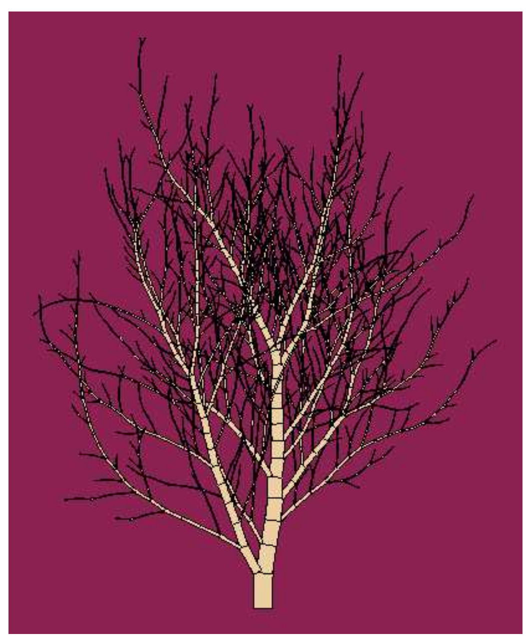


Continued fraction trie



Weyl tree by Devroye, versus 'beta tree'





The Saga of Digital Trees

3. Data Bases

Adaptive hashing schemes

Tries are very versatile.
— They can be paginated (bucketted): stop splitting at b.
— They can be combined with *hashing* to cope with non-uniformity of data.

Near 1977-78, several groups discover the virtues of dynamic hashing. Idea: *Split buckets instead of chaining them.* (Larson; Fagin-Nievergelt-Pippenger-Strong; Litwin)

Expected size of b–tree is $\frac{n}{b \log 2} + \text{fluctu}$, corresponding to 69% filling ratio.

Compare with similar ratio for B–trees (Yao)

2 accesses suffice for very large DB.

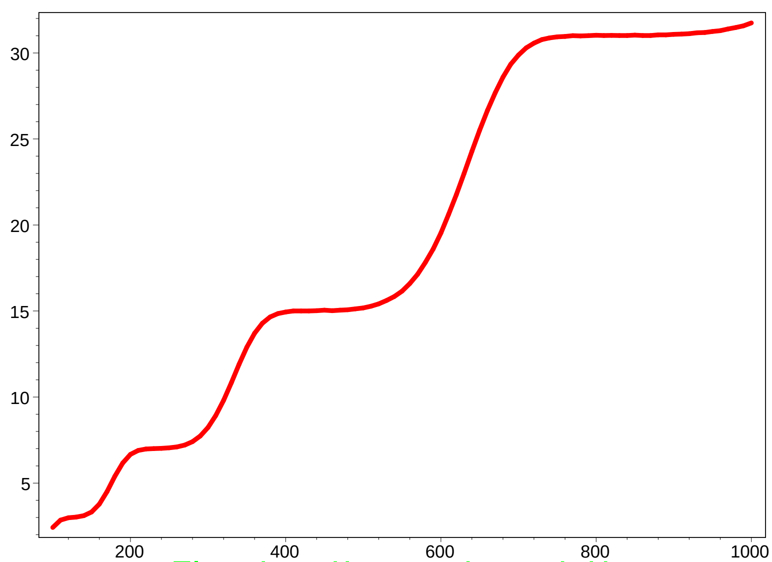
Extendible Hashing transforms the index into a perfect tree $\equiv$ array that can be paginated.

H=height   Index size $\equiv 2^H$

(Yao, Régnier, F., ca 1980)

$$\mathbb{E}(H) \sim \left(1+\frac{1}{b}\right)\log_2 n; \qquad \mathbb{E}(2^H) \approx 4^{\text{fluctu}} n^{1+1/b}.$$

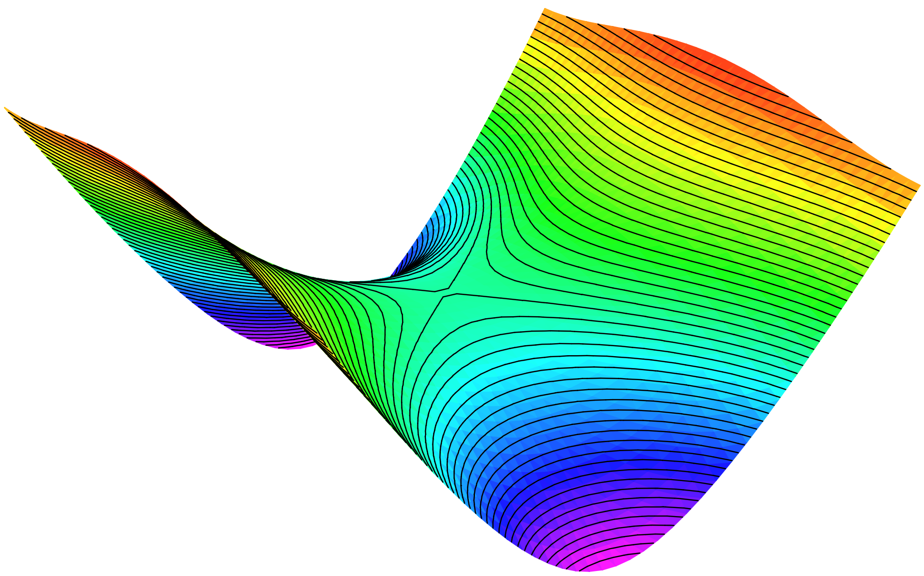
$(b = 50)$



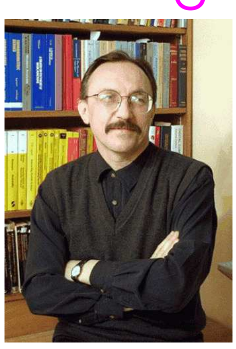
Fluctuations *do* exist!

Height: One of the very first intrusions of saddle point method in Analysis of Algorithms.

$$[z^n]f(z) = \frac{1}{2i\pi}\oint f(z)\,\frac{dz}{z^{n+1}}.$$



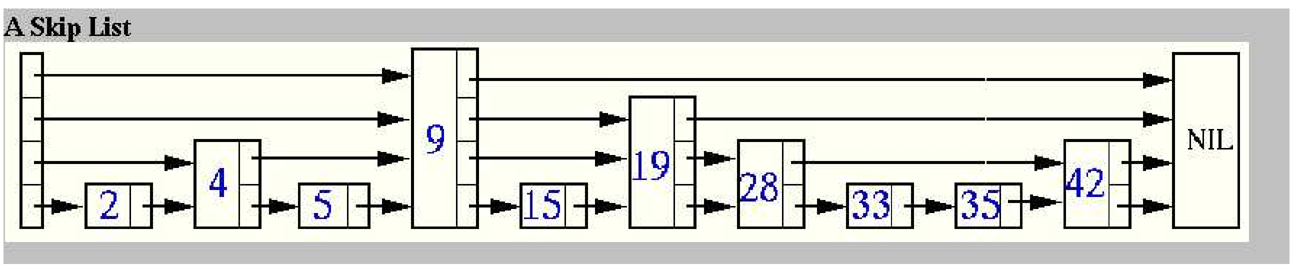
$\rightsquigarrow$ Jacquet & Szpankowski's "analytic de-Poissonization": analyse under probabilistic model with "imaginary probabilities"!



Skip lists

From VSAM's to skip lists

Idea 1 (old): build indexes of indexes of indexes . . .
Idea 2: balance $\rightsquigarrow$ B–trees
Idea 2′: randomize! $\equiv$ Pugh's skip lists



Much easier to maintain than balanced structures!

Analysis by Papadakis + Munro, Poblete
Kirschenhofer, Martínez, Prodinger entirely based on trie technology.

Probabilistic counting algorithms

Can you estimate to 5% the number of different words in Shakespeare given a pencil and one sheet of paper?

Yes. F.+Martin (1985) for data base query optimization.

Ideas: hash to get uniformity; observe bit patterns.
`0...` = 50% of times; `10...` = 25%; `110...` = 12.5%

Try 2^K where $\overbrace{11\cdots1}^{K}0\ldots$ is longest initial run of 1's.
The best observable known is trie-like and has accuracy

$$\frac{\boxed{0.78}}{\sqrt{m}} \qquad \text{for } m \text{ words of memory}$$

(+"stochastic averaging"); $\boxed{0.78}$ is a Mellin constant.

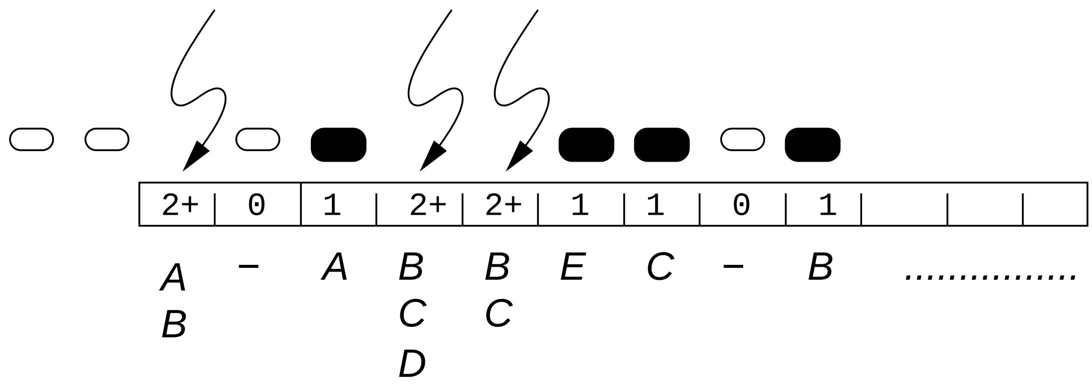
Works in distributed environment:
Yellow pages of New York ∪ San Francisco by phone line!

Data mining applications. Quick running counts in routers (Durand 2003) based on other trie observables.

The Saga of Digital Trees

3. Protocols

♡ 1970: the shared communication channel

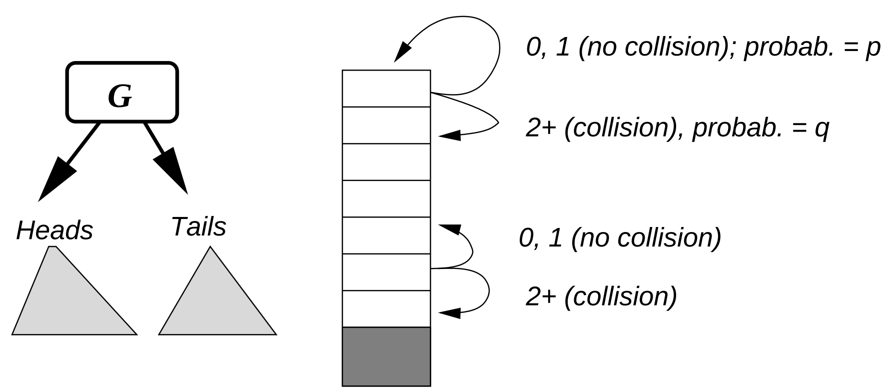


Ethernet: Try; wait $\times 1$, $\times 2$, $\times 4$, etc

$\rightsquigarrow$ Aldous 1987: *Ethernet is unstable!*

♡ 1977: The **Tree/Stack protocol**

CTM = Capetanakis, Tsybakov, Mikhailov



= A digital trie but with a flow of arrivals!

♠♠ Erroneous analyses missed the wobbles.

Variance by Kirschenhofer, Prodinger et al.= Mellin + modular forms.

Tree protocol $\Longrightarrow$ Poisson GenFun solves $(p+q=1)$

$$\psi(z)-\psi(\lambda+pz)-\psi(\lambda+qz)=\text{toll}\,(z).$$

A non-commutative iteration semigroup with a globally invariant measure.

Theorem. Stable till $\lambda_{\max}=0.36017$ root of:

$$-\frac{1}{2} = \frac{e^{-2x}}{1-2x}\sum_{i\geq 0} 2^i g\left(\frac{x}{2^i}\right),$$
$$g(y) := e^{-2y}\left((e^{-y}(1-y)-1+2y(1+y))\right).$$

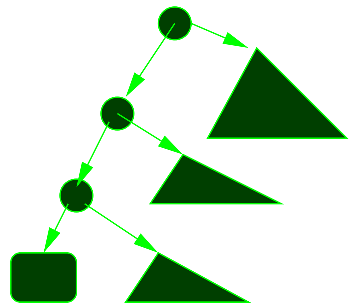
Analyses by Fayolle, F., Hofri, Jacquet, Mathys $\Longrightarrow$
— Ternary tree algorithms gives 10% better throughput
— Protocol is hyperstable at all arrival rates.

The IEEE 802.14 norm...a *failed* success story!

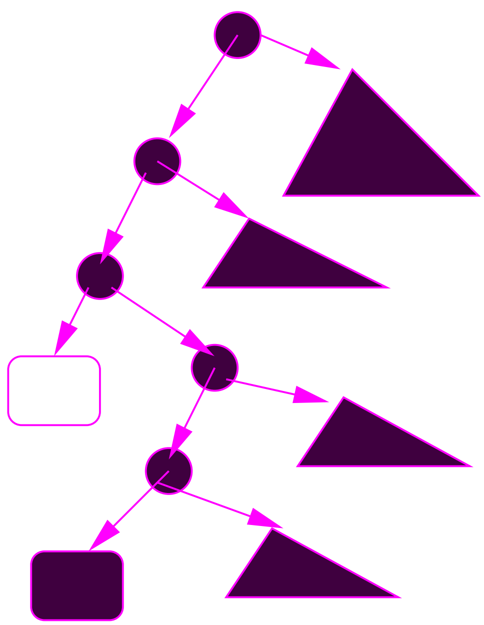
Also analyses by Greenberg+F+Ladner: tree protocol modified to attain 93% of optimal: $\lambda_{\max}=0.4672$.

Leader Election:

(i) The leftmost branch of a trie



(ii) The leftmost border of a trie



Analyses by Fill, Mahmoud, Szpankowski, Prodinger, F+Sedgewick; includes distributions.

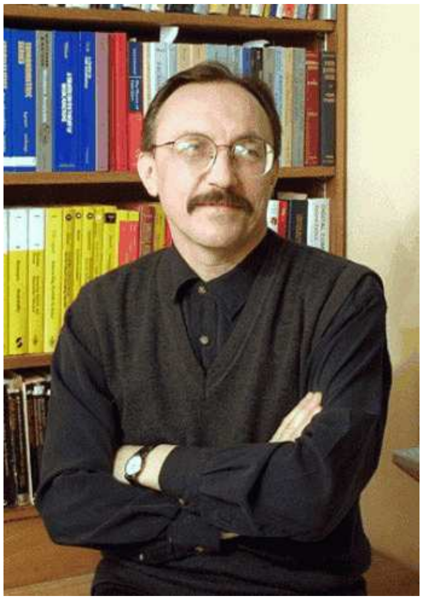
$(2 \log n)$ rounds; $\log_2 n$ rounds.

The Saga of Digital Trees

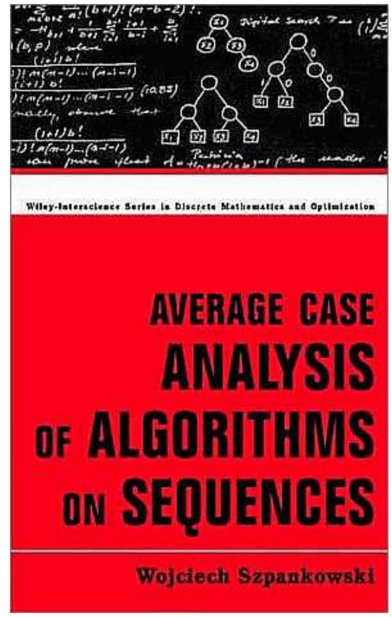
4. Text and compression

Tries meet texts again!

Szpankowski's *Analysis of Algorithms on Sequences*.



$\leadsto$



Random text: `kwnbpr hwnqqcpq yt nxgfhsd agghos fhskla zmmxnz kasiweyzkcn ejhjsal ehrdjn...`

$\neq$ "Natural" language text: *Cale Pismo przez Boga jest natchnione i pozyteczne do nauki, do wykrywania bledow...*

Can be compressed!

- *Lempel & Ziv* invent LZ *compression* (1977+) based on building adaptive dictionaries.

a|b|r|ac|ad|ab|ra|abr|aca|d|abra|abrac|ada|br|aa|br|acad|abraa|..

Turns out to be related to *digital search trees*.

• Régnier-Jacquet (1987) do *distributional analysis* of tries under Bernoulli models.

• Szpankowski-Jacquet (1990) do average-case analysis of tries under *Markovian dependencies*.

• Jacquet–Szpankowski–Louchard (1995+) extend distributional analysis to DST's:

$$\boxed{\frac{\partial}{\partial z}F(z,u) = F(z,pu)F(z,qu) + \text{fudge}}$$

$\rightsquigarrow$ Combines everything:
algebra of trie costs, Mellin, analytic dePoissonization. . .

♡♡ Complete characterizations of *Lempel-Ziv* algorithms, notably: redundancy.

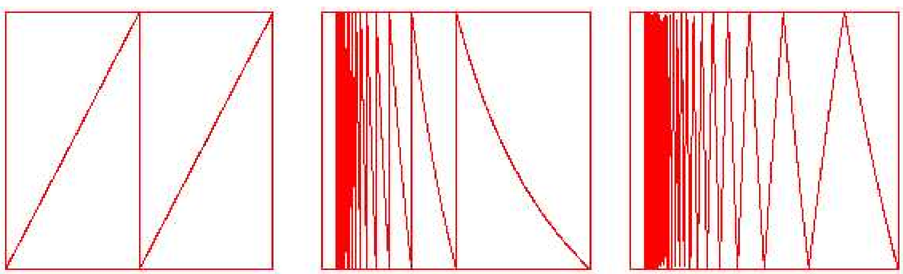
The Saga of Digital Trees

5. Geometry & Dynamical Systems

• "Thermodynamic formalism" by Ruelle (1970)
• Operators & Euclid's alg. by Babenko, D. Mayer (1977.)
• Related to information theory & tries by Vallée (1995+)

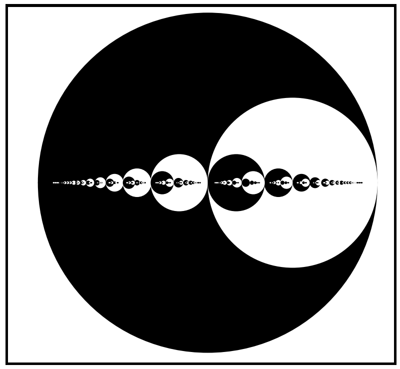
T is a transformation. Iterates?



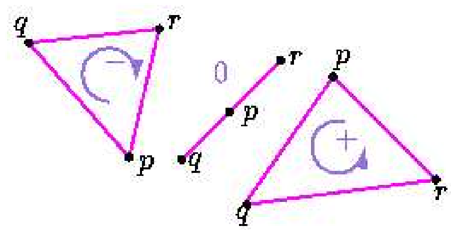
Transfer operator: $\mathcal{G}_s[f](x) := \sum_{h \in T^{-1}} (h'(x))^s f \circ h(x).$

Vallée: Spectra & functional analysis serve to generate probabilities of prefixes $\rightsquigarrow$ tries.

Tries under dynamic source models;
$\rightsquigarrow$ Bentley-Sedgewick's Ternary Search Tries
Entropy for size, depth path-length; Eigenvalue $\lambda(2)$ for height, etc.



Applies to continued fraction representations & algs:
$\rightsquigarrow$ HAKMEM Algorithm (Gosper, 1972); 2D orientation = Avnaim, Boissonnat, Devillers, Preparata, Yvinec 1997.



$$\frac{36}{113} = \cfrac{1}{3 + \cfrac{1}{7 + \cfrac{1}{\mathbf{5}}}}, \qquad \frac{113}{355} = \cfrac{1}{3 + \cfrac{1}{7 + \cfrac{1}{\mathbf{16}}}}.$$

$\rightsquigarrow$ Sorting with continued fractions, cost:

$$K_0 n \log n + K_1 n + Q(n) + K_2 + o(1),$$

$$K_0 = \frac{6 \log 2}{\pi^2}, \quad K_1 = 18 \frac{\gamma \log 2}{\pi^2} + 9 \frac{(\log 2)^2}{\pi^2} - 72 \frac{\log 2 \zeta'(2)}{\pi^4} - \frac{1}{2}.$$

♡♡♡ Q depends on Riemann hypothesis!!!

The Saga of Digital Trees

6. Everywhere. . .

Random Trie Encounters

♡ Polynomial factorization (Cantor-Z refinement)

```
> factor(x^13-x^10+x^5-x^2+x^3-1);
                2              6    4         2         2
 (x - 1) (x  - x + 1) (x  - x  + 1) (x  + x + 1)
```

Vol 2., F+Gourdon+Panario

♡ Quadtries and geometry, multiD search
Rivest–Bentley–Samet

♡ Other probabilistic counting algorithms
Morris–Freivalds, Wegner's, etc

♡ Binary Decision Diagrams (BDD's) by Bryant (!?)
= Fully developed tries + common subtree factoring...

♡ Hierarchical data compression by J. Kieffer

♡ Level compressed tries $\Longrightarrow$ fast lookup in routers!
Nilsson et al.

Finally . . .

Where are we?

Analysis of algorithms as of now:

Complex Models

. . . become more and more tractable.

♡ A large number of basic algorithms have been analysed. Cf Sedgewick's book.

♡ Symbolic Methods help translate complex probabilistic models into gen. functions.

♡ Analytic Combinatorics = an extensive calculus of asymptotic properties based on singularities.

⇝ A unified theory of basic random combinatorial structures and algorithms.

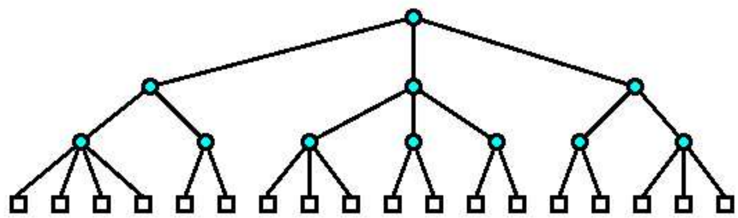
♡ Fruitful connections with computer algebra.

⇝ Automatic counting, automatic asymptotics, automatic random generation.

Two basic principles $\mapsto$ "dictionaries"

SYMBOLIC METHODS

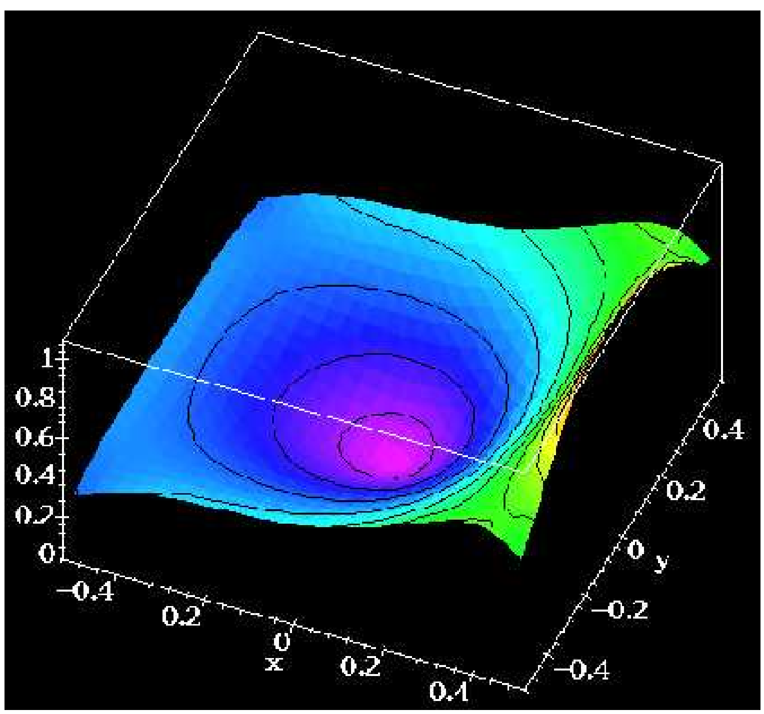
Generating functions

 $\mapsto$ $\mathbf{z}^{11}$

$$\mathbf{z} + \mathbf{z}^2 + \mathbf{z}^3 + 2\,\mathbf{z}^4 + 2\,\mathbf{z}^5 + 4\,\mathbf{z}^6 + 5\,\mathbf{z}^7 + 9\,\mathbf{z}^8 + \cdots$$

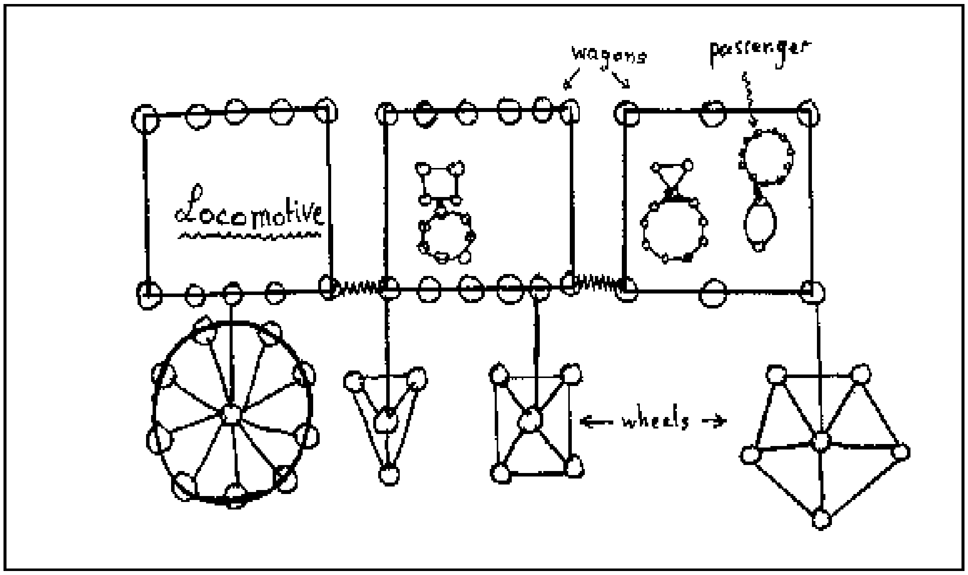
$$f(z) = z + f(z^2 + z^3 + z^4)$$

ANALYTIC FUNCTIONS AND SINGULARITIES



The example of TRAINS

- Cope with complex structural "specifications"



train.mws – [Server 1]

```
> with(combstruct);
        [allstructs, count, draw, finished, gfeqns, gfseries, gfsolve, iterstructs, nextstruct]
> TRAINS:={T=Prod(Loco,Sequence(Wagon)),
     Wagon=Prod(Loco,Set(Passenger)),
     Loco=Sequence(Prod(Board,Wheel),card>=1),
  Passenger=Prod(Head,Belly),
     Board=Prod(Z,Z), Wheel=Prod(Z,Cycle(Z)),
     Head=Cycle(Z), Belly=Cycle(Z)}:
> seq(count([T,TRAINS,labelled],size=n),n=0..20);
0, 0, 0, 0, 24, 60, 240, 1260, 88704, 786240, 10800000, 151351200, 4295047680, 77837760000,
   1696330097280, 37148330419200, 1032008553354240, 26762644285977600,
   759278023249374720, 22080501641848366080, 710352165477065472000
> subs(gfsolve(TRAINS,labelled,z),T(z));
```

$$-\frac{z^3 \ln\left(-\frac{1}{-1+z}\right)}{-1 + z^3 \ln\left(-\frac{1}{-1+z}\right) + z^3 \ln\left(-\frac{1}{-1+z}\right) e^{\left(\ln\left(-\frac{1}{-1+z}\right)^2\right)}}$$

$$0.1008557594\ (0.5180547070)^{(-\ n)} + \ldots$$

Analytic Combinatorics

= organize *random discrete structures* (cf. stochastic proc.)
= tightly coupled with Analysis of algs.

- Permutations: order stat., search & sort.
- Words: patterns, comput. biology, coding
- DIGITAL TREES
- Allocations: hashing, comb. opt., . . .
- Graphs: combinat opt., networks (?)
- Trees: symbolic manipulation, etc.

THERE IS A story about two friends, who were classmates in high school, talking about their jobs. One of them became a statistician . . . "And what is this symbol here?" "Oh," said the statistician, "this is pi." "What is that?" "The ratio of the circumference of the circle to its diameter." "Well, now you are pushing your joke too far," said the classmate, "surely the population has nothing to do with the circumference of the circle."

The miracle of the appropriateness of the language of mathematics for the formulation of the laws of physics is a wonderful gift which we neither understand nor deserve. We should be grateful for it and hope that it will remain valid in future research and that it will extend, for better or for worse, to our pleasure, even though perhaps also to our bafflement, to wide branches of learning.

— Eugene Wigner

"The Unreasonable Effectiveness of Mathematics in the Natural Sciences," in Communications in Pure and Applied Mathematics, vol. 13, No. I (February 1960).