BEYOND POPULAR SCIENCE

DAVID H. SILVER

BEYOND POPULAR SCIENCE
David H. Silver

https://www.openbookpublishers.com

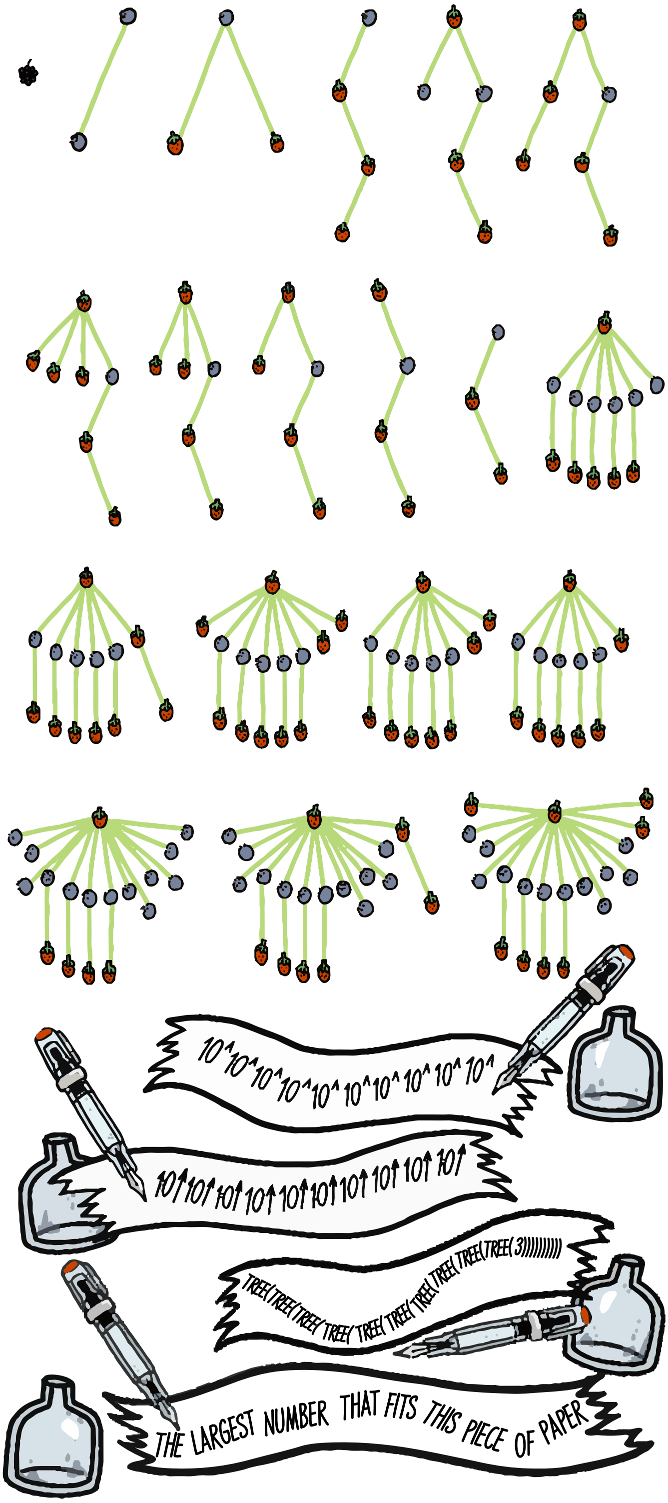
> David H. Silver, *Beyond Popular Science*. Cambridge, UK: Open Book Publishers, 2026, `https://doi.org/10.11647/OBP.0526`

Digital material and resources associated with this volume are available at
`https://doi.org/10.11647/OBP.0526#resources`

ISBN Paperback: 978-1-80511-877-0
ISBN Hardback: 978-1-80511-878-7
ISBN Digital (PDF): 978-1-80511-879-4
ISBN HTML: 978-1-80511-881-7
ISBN Digital ebook (epub): 978-1-80511-880-0
DOI: 10.11647/OBP.0526

Cover image by Enny Silver and David H. Silver
Cover design by Jeevanjot Kaur Nagpal

The Busy Beaver That Ate the TREE



Top (Valid TREE(3) Sequence Prefix): An initial segment of a TREE(3)-avoiding sequence. Each rooted tree is added in such a way that no earlier tree is homeomorphically embeddable into a later one. The sequence starts with simple trees and becomes rapidly more complex. Exponentiating a googolplex by itself a googolplex times is comically dwarfed by TREE(3). **Bottom (Largest Number Competition):** Using 20 cm of ink, write the largest number possible. Early entries include exponential towers and Knuth up-arrow notation. But one leverages the TREE function hierarchy in TREE(TREE(3))—a number vastly larger than previous contenders. The final banner is a different type of entry, more reminiscent of Rayo’s number.

The Busy Beaver That Ate the TREE

Imagine you are given a pen with enough ink to write 20 centimetres and you are to write the biggest number you can think of. You can start by a tower of exponents $10^{10^{\cdots^{10}}}$, but it is not that big. In this chapter we explore the hierarchy from computable functions to TREE(3)—a number so immense that even if you built a tower of exponentials starting with a trillion raised to the power of a trillion, and then repeated that construction every attosecond for a trillion years, the result would still be vanishingly small in comparison. Yet even TREE(3) sits as close to infinity as the number eight, placing us in an infinity zoo where sizes exceed the categories brains evolved to handle.

Sand Reckoner ∘ Tower Notation ∘ Knuth Arrows ∘ 20cm Ink Competition ∘ Ackermann Function ∘ TREE(3) Combinatorics ∘ Tree Embedding Game ∘ Busy Beaver ∘ Halting Undecidability ∘ Rayo's Function ∘ Large Cardinals

„Das Unendliche hat wie keine andere Frage
von jeher so tief das Gemüt der Menschen bewegt…
Aus dem Paradies, das Cantor uns geschaffen,
soll uns niemand vertreiben können.“

(“The infinite! No other question has ever moved so profoundly the spirit of man… Cantor's paradise, from which no one will expel us.”)

— David Hilbert, 1926

The Busy Beaver That Ate the TREE

> The challenge of *expressing vast quantities* appears across ancient texts. The Hebrew Bible uses ‘רִבֵּי רְבָבוֹת’ (*ribei revavot*)—myriads of myriads—to denote numbers beyond ordinary counting, as in Daniel 7:10 describing the heavenly host: ‘thousand thousands ministered unto him, and ten thousand times ten thousand stood before him.’ This poetic multiplication hinted at systematic ways to build larger numbers.
>
> Archimedes formalised this intuition in *The Sand Reckoner* (Ψαμμίτης, Psammites), written circa the early third century BCE as a letter to Gelon, son of King Hiero II of Syracuse. The work addressed Aristarchus’s heliocentric model, which implied a universe vastly larger than previously imagined. Archimedes asked: could one count the grains of sand needed to fill such a cosmos?
>
> Greek numerals stopped at a myriad (10,000). Archimedes extended them by defining ‘orders’—the first order contained numbers up to a myriad myriads (10^8), the second order began at 10^8 and continued to $(10^8)^2$, and so forth. Using this system, he estimated the universe could hold at most 10^{63} sand grains. The calculation was secondary to the method: *The Showing* that any finite quantity, however vast, could be expressed and manipulated. This was a milestone in scientific notation and the separation of numbers from physical counting.
>
> Edward Kasner popularised the terms after asking his nine-year-old nephew, Milton Sirotta, to name 10^{100}; Milton proposed ‘googol,’ and they defined ‘googolplex’ as 10^{googol}. The coinage predates the book, but the terms were widely disseminated in Kasner and Newman’s *Mathematics and the Imagination* (1940), illustrating how notation can grow rapidly.
>
> Modern developments began with Wilhelm Ackermann’s 1928 function that grows faster than any primitive recursive function. This gave rise to the computational growth rates forming a hierarchy—some functions outpace others so dramatically that conventional notation fails.
>
> Harvey Friedman migrated large numbers from recreational mathematics into serious research in the 1990s. His TREE sequence, derived from Kruskal’s tree theorem, produced numbers dwarfing all previous constructions. TREE(3) is finite but so large it cannot be expressed using conventional operations, iterated any reasonable number of times. The proof requires axioms beyond Peano arithmetic, connecting large numbers to results in proof theory and the limits of formal systems.

Archimedes wrote the *Sand Reckoner* to count sand grains in the cosmos. His real purpose was notation—showing how large numbers could be handled by grouping units into ‘orders’ and assigning names to powers of powers. This marked one of the first recorded attempts to handle orders of magnitude through notation.

Children discover this principle through play. Counting on fingers reaches ten. Tally marks extend to dozens. Roman numerals handle thousands awkwardly. Arabic numerals with positional notation reach millions effortlessly.

https://doi.org/10.11647/OBP.0526.17

The number 10^{10}—ten billion—sits at the edge of intuitive grasp. Demographers estimate that about 10^{11} humans have ever lived. The observable universe contains approximately 10^{80} atoms. Scientific notation makes this tractable: '1 followed by 80 zeros.' But even this notation meets its limits.

Consider $10^{10^{10}}$. This number has ten billion digits. If each digit were an atom, we would need 10^{10} atoms to write it down—an incomprehensibly tiny fraction of the universe's 10^{80} atoms (10^{-70} of them). Yet the notation remains compact—just a few symbols capture a magnitude that dwarfs physical representation.

Now build a tower of many years: $10^{10^{10^{\cdot^{\cdot}}}}$ with ten tens, call it T_{years}. To grasp this scale, imagine beings who create universes by setting random initial conditions, attempting to arrange cosmic evolution so that 13.8 billion years later, if a civilisation emerges on the resulting Earth, they launch exactly 100 trillion green peas toward the Moon, and all land in a pre-specified bucket at exactly 8:00:00.00000 PM on Friday, 4 July. The initial conditions must account for every quantum fluctuation, every gravitational interaction, the formation of galaxies, stars, planets, the evolution of life, agriculture, spaceflight technology, and the precise timing of launch.

Most attempts fail—Earth never forms, or forms without life, or life evolves differently, or the civilisation launches peas a second too early. When an attempt fails, they wait 10^{100} years until that universe reaches heat death, then start fresh with new initial conditions. To achieve a billion consecutive successes will require much less time than T_{years}.

Physical metaphors lose meaning at these scales. No arrangement of atoms, no duration of time, no cosmic process captures numbers this large. We can develop formal notation that builds recursively, where each operation multiplies growth rates.

Knuth's arrow notation compresses this tower-building (Knuth, 1974). One arrow denotes exponentiation, $10 \uparrow 10 = 10^{10}$. Two arrows build a tower, $10 \uparrow\uparrow 10 = 10^{10^{10^{\cdot^{\cdot}}}}$ with ten 10s. Three arrows iterate the two-arrow operation, $10 \uparrow\uparrow\uparrow 10 = 10 \uparrow\uparrow (10 \uparrow\uparrow\uparrow 9)$, building towers of towers recursively. Each arrow multiplies the growth rate beyond comprehension.

Frame this as a competition. You have ink for 20 centimetres of writing. Produce the largest number possible. Every symbol must be precisely defined. Writing '10^{10}' beats '10000000000'—notation outpaces digits. With Knuth's arrows, $3 \uparrow 3 = 3^3 = 27$, but $3 \uparrow\uparrow 3 = 3^{3^3} = 3^{27}$, already over 7 trillion. Now, instead of manually building recursive stacks, we can define functions that generate them.

The Ackermann function grows faster than any primitive recursive function:

$$\begin{aligned} A(0,n) &= n+1 \\ A(m+1,0) &= A(m,1) \\ A(m+1,n+1) &= A(m,A(m+1,n)) \end{aligned}.$$

Starting modestly—$A(1,n) = n+2$, $A(2,n) = 2n+3$, $A(3,n) = 2^{n+3}-3$—by $A(4,2)$ we reach $2^{65536}-3$, computed from a power tower five levels high, $2^{2^{2^{2^{2}}}}-3$. A well-chosen function reference like '$A(A(10,10), A(10,10))$' beats unfathomable explicit digits. The best use of ink is no longer to write numbers, but to specify methods of generation.

Beyond recursive towers lies combinatorial explosion. While Ackermann and arrows build through iteration, TREE(3) emerges from a simple game with trees that generates growth surpassing any tower of exponentials. The leap from arithmetic to combinatorics produces numbers that dwarf all previous constructions.

Take a deep breath and play a game called TREE(n). You draw a sequence (Kruskal, 1960) of trees—not botanical trees, but branching diagrams with a single root at top, branches splitting downward, with vertex colours from an n-colour set. You have n colours available (say, red, blue, and green for TREE(3)). The rules: the i-th tree in your sequence can have at most i vertices; no earlier tree may embed in any later tree.

An earlier tree embeds in a later tree if there is an injective, colour-preserving map of vertices that preserves lowest-common-ancestor relations. In simpler terms, tree A embeds in tree B if you can find a subset of B's vertices that matches A's structure and colours exactly.

TREE(1) = 1. With one colour (say, only red), you draw a single red vertex. The second tree needs two vertices, so it must have two red vertices—but any configuration of two red vertices embeds the single red vertex. Game over.

TREE(2) = 3. With two colours (red and blue), the longest sequence is (1) single red vertex, (2) blue root with blue child below, (3) single blue vertex. Try adding a fourth tree with at most 4 vertices using red and blue—any configuration will embed one of these three.

TREE(3) is where the magnitude explodes. This number dwarfs any fixed-height tower of exponentials and values produced by many natural hierarchies at modest inputs. However, the Ackermann function eventually exceeds any fixed bound for sufficiently large inputs. If every atom in the observable universe became a tower of googolplexes, and these googolplex-atoms multiplied together every nanosecond throughout cosmic history, the result wouldn't approach one trillionth of TREE(3).

TREE(3) is a specific, well-defined number. There exists a definite answer to "what is the 97th digit of TREE(3)?" We simply cannot compute it. Enter functions that grow even faster through different mechanisms. The next numbers are more dependent on the underlying axioms and the language used to define them.

TREE arises from combinatorial constraints—avoiding embeddings in sequences. The busy beaver function BB(n) shifts from combinatorial to computational (Radó, 1962) limits. Among all n-state Turing machines (theoretical computing devices) that eventually halt, BB(n) equals the maximum number of steps any such machine takes before stopping. The known values are BB(1)=1, BB(2)=6, BB(3)=21, BB(4)=107. For BB(5), the value is known to be $47{,}176{,}870$. BB(6) exceeds $10 \uparrow\uparrow 15$.

Unlike TREE(3), BB(n) derives its magnitude from undecidability. No algorithm can compute BB(n) for arbitrary n, as this would solve the halting problem (Turing, 1936)—proven impossible by Turing. The function eventually surpasses TREE(n) because it encompasses all computational processes, including those calculating TREE values. Recent analysis suggests BB(2645) likely exceeds TREE(3), and from then on, it grows explosively faster.

The ultimate strategy abandons specific constructions for meta-mathematical limits. Rather than defining a particular fast-growing function, we can ask: what is the largest number definable within the language itself?

Rayo's function does that while venturing into linguistic and logic territory. At MIT's 2007 "Big Number Duel," philosopher Agustín Rayo proposed the ultimate strategy—define Rayo(n) as the largest natural number expressible in first-order set theory using at most n symbols. This approaches the theoretical limit of our 20 cm game—essentially encoding "the largest number definable with this much notation" within formal logic.

Rayo(n) outgrows any function definable in its own language through diagonalization—exceeding every possible definition. It dwarfs both TREE(n) and BB(n).

All these finite numbers—from towers of exponentials to TREE(3) to Rayo's function—remain infinitely far from infinity. They demonstrate ingenuity in naming ever-larger quantities, yet each sits at the same infinite distance from the first infinity.

And so, we now turn to infinity. When we cross to infinity, the rules of growth change. Infinity comes in two flavours—cardinals (sizes of sets) and ordinals (positions in well-ordered sequences). Think of 'three' (counting objects) versus 'third' (position in line). The smallest infinite cardinal is $\aleph_0$ (aleph-null), the cardinality of natural numbers. The smallest infinite ordinal is ω, their order type.

Cardinal arithmetic is different from finite arithmetic: $\aleph_0 + 1 = \aleph_0$—Hilbert's Hotel has infinitely many rooms (Hilbert, 1926), all full, yet can accommodate one more guest; $\aleph_0 + \aleph_0 = \aleph_0$—interleave odds and evens; $\aleph_0 \times \aleph_0 = \aleph_0$—arrange rationals in a grid.

But exponentiation breaks this pattern. ${\aleph_0}^{\aleph_0} > \aleph_0$. Exponentiation represents functions between sets. In finite cases, $5^5 = 3125$ is exactly the number of possible functions from $\{1, 2, 3, 4, 5\}$ to itself. Similarly, ${\aleph_0}^{\aleph_0}$ represents all functions from naturals to naturals, yielding the continuum's cardinality which is larger than $\aleph_0$.

With ordinals, exponentiation truly explodes. Form ω^ω—omega to the omega power. Then ω^{ω^ω}—a tower of omegas. But why stop at finite towers? We're already working with infinity! Build an infinite tower: $\omega^{\omega^{\omega^{\cdot^{\cdot^{\cdot}}}}}$ with ω many ω's. This is the limit of finite towers, well-defined in ordinal arithmetic. This mind-bending construction yields ε_0, the first epsilon number, satisfying $\omega^{\varepsilon_0} = \varepsilon_0$.

This unimaginably large ordinal, built from an infinite tower of infinities, is tiny in the hierarchy of infinities. It's merely the first in a new regime: ε_1 is the next fixed point after ε_0; ε_ω is the ω-th fixed point; $\varepsilon_{\varepsilon_0}$ uses our 'massive' infinity as a mere index.

How do we organise these ever-larger infinities? In 1908, mathematician Oswald Veblen developed a hierarchy. Start with the function $\varphi_0(\alpha) = \omega^\alpha$—this generates our familiar exponential towers. The function φ_1 then enumerates all the epsilon numbers (those fixed points where $\omega^x = x$). The function φ_2 finds all the fixed points of φ_1—ordinals so large that even the epsilon-generating function cannot reach them. Each level finds what the previous level missed, climbing an infinite ladder where each rung reveals new unreachable ordinals above.

This process continues through all finite indices: φ_3, φ_4, and onward. But now we can define φ_ω, then $\varphi_{\omega+1}$, even $\varphi_{\varphi_0(0)}$. The indices themselves become ordinals! Eventually, we reach an ordinal so large it equals its own index in the Veblen hierarchy, $\Gamma_0 = \varphi_{\Gamma_0}(0)$. This is the Feferman-Schütte ordinal, discovered independently by Solomon Feferman and Kurt Schütte in the 1960s.

Γ_0 marks more than just another large ordinal. Below Γ_0, we can build ordinals step by step using explicit rules. Beyond it, we need new principles. In technical terms, Γ_0 is the proof-theoretic ordinal of predicative mathematics—it measures exactly how far we can count using only definitions that refer to previously constructed objects. To go further requires impredicative methods: definitions that refer to totalities containing the very object being defined. It's like trying to lift yourself by your own bootstraps—impossible in physics, but sometimes necessary in mathematics. Beyond Γ_0 lie ordinals and cardinals requiring ever-stronger principles.

A word of caution—beyond this point (and a little bit before this point to be honest), we enter territory inhabited almost exclusively by logicians and set theorists. These larger ordinals and cardinals, while mathematically precise, have little connection to anything outside specialised logical discussions. They represent abstract possibilities rather than quantities that arise naturally in mathematics. Yet surprises occur—just as TREE(3) emerged from combinatorics to dwarf all previous numbers, these abstract ordinals occasionally appear in analysis. Still, for most purposes, this glimpse into the hierarchy suffices.

ω_1^{CK} (Church-Kleene ω_1)—the first ordinal with no computable description. Every ordinal before this can be described by some computer program, even if that program would run forever. But ω_1^{CK} transcends computation itself. No algorithm, no matter how clever, can specify this ordinal.

ω_1—the first uncountable ordinal. All ordinals before this can be put in one-to-one correspondence with natural numbers. But ω_1 is the first ordinal too large for any such pairing. If you tried to list all smaller ordinals as first, second, third..., you would run out of natural numbers before reaching ω_1. It's a bigger kind of infinity.

Inaccessible cardinals—infinite numbers unreachable by standard set operations. You cannot reach an inaccessible cardinal by taking powers (such as $2^{\aleph_0}$), unions, or any combination of usual set-theoretic operations starting from smaller cardinals.

Measurable cardinals—infinite numbers large enough to support probability measures. On finite sets, we can assign probabilities—half the integers from 1 to 10 are odd. But on infinite sets, this usually fails. Measurable cardinals are so large that probability makes sense again. Beyond these lie supercompact cardinals, where the entire universe of sets at any level looks like a small-scale model of the universe at the supercompact level.

Hierarchy of Growth Rates

Level 1: Elementary Functions

$$f(n) = n + c \text{ (linear)}$$
$$f(n) = n^k \text{ (polynomial)}$$
$$f(n) = k^n \text{ (exponential)}$$
$$f(n) = n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$$

Level 2: Iterated Exponentials

Tetration: ${}^k a = \underbrace{a^{a^{\cdot^{\cdot^{a}}}}}_{k \text{ times}}$

Digits of ${}^k 2$ are about $(\log_{10} 2) \cdot {}^{k-1}2$.

Level 3: Ackermann Function

$$A(0, n) = n + 1$$
$$A(m + 1, 0) = A(m, 1)$$
$$A(m + 1, n + 1) = A(m, A(m + 1, n)).$$

Growth: $A(1, n) = n + 2$, $A(2, n) = 2n + 3$, $A(3, n) = 2^{n+3} - 3$, $A(4, n) = \underbrace{2^{2^{\cdot^{\cdot^{2}}}}}_{n+3} - 3$.

Level 4: Knuth Arrows

$$a \uparrow^1 b = a^b$$
$$a \uparrow^n 1 = a \text{ for } n \geq 1$$
$$a \uparrow^n 0 = 1 \text{ for } n \geq 1$$
$$a \uparrow^n b = a \uparrow^{n-1} (a \uparrow^n (b - 1))$$
$$\text{for } n \geq 1, b > 1.$$

Extension: $a \uparrow^0 b := ab$.
$3 \uparrow 3 = 27$, $3 \uparrow\uparrow 3 = 7{,}625{,}597{,}484{,}987$
$3 \uparrow\uparrow\uparrow 3 = 3 \uparrow\uparrow 7{,}625{,}597{,}484{,}987$

Level 5: Fast-Growing Hierarchy

Indexed by ordinals:

$$f_0(n) = n + 1$$
$$f_{\alpha+1}(n) = f_\alpha^n(n)$$
$$f_\lambda(n) = f_{\lambda[n]}(n) \text{ for limit } \lambda$$
$$f_\omega(n) = f_n(n)$$
$$f_{\omega^2}(n) = f_{\omega \cdot n}(n)$$
$$f_{\varepsilon_0}(n) \text{ dominates finite } \omega \text{ towers}$$

Level 6: TREE Function

TREE(n) = max sequence of rooted finite trees with vertices coloured from an n-element set; on step i the tree has at most i vertices; forbid homeomorphic embedding from any earlier tree into any later.
TREE$(1) = 1$, TREE$(2) = 3$
Via Kruskal's theorem, associated length functions dominate f_α for all $\alpha < \theta(\Omega^\omega)$; TREE$(3)$ is far beyond f_{ε_0}-scale growth.

Level 7: Busy Beaver

BB(n) = max steps of any halting n-state, 2-symbol TM.
BB$(4) = 107$, BB$(5) = 47{,}176{,}870$ (proven 2024)
BB$(6) > 10 \uparrow\uparrow 15$ (lower bound)
Eventually dominates all computable functions.

Level 8: Rayo Function

Rayo(n) = the least natural number greater than every number definable in first-order set theory by a formula of length $\leq n$ (with fixed encoding).
Dominates any n-symbol definable function by diagonalization.

Growth Comparison

For large n: polynomial $\ll$ exponential $\ll$ Ackermann $\ll$ arrows $\ll$ f_{ε_0} $\ll$ TREE $\ll$ BB $\ll$ Rayo

Each level uses fundamentally stronger recursion principles. Comparison depends on proof-theoretic bounds (FGH, Kruskal), computability (BB), and definability (Rayo).

References:

M. H. Löb and S. S. Wainer, ‘Hierarchies of number-theoretic functions I/II,’ Archiv für mathematische Logik und Grundlagenforschung (1970–72).
H. Friedman, ‘Finite forms of Kruskal's theorem,’ Journal of Combinatorial Theory, Series A **95**, 102–144 (2001).