DAWN PUTNAM SESSION

Putnam 2020 — Papers A and B

Putnam 120 Archive

Contents

Putnam 2020 — Paper A **5**
2020 A1 5
2020 A2 7
2020 A3 9
2020 A4 11
2020 A5 15
2020 A6 20

Putnam 2020 — Paper B **25**
2020 B1 25
2020 B2 27
2020 B3 30
2020 B4 33
2020 B5 37
2020 B6 42

Putnam 2020 — Paper A

2020 A1

2020 A1. How many positive integers N satisfy all of the following three conditions? (i) N is divisible by 2020. (ii) N has at most 2020 decimal digits. (iii) The decimal digits of N are a string of consecutive ones followed by a string of consecutive zeros.

Fields: number theory (divisibility, orders mod p) **Position:** A1 – opener, but the order-of-10 step is a genuine idea, not arithmetic **Answer:** 508536.

Layer 1 THE WHAT — surface analysis

The natural attempt. The description in (iii) is concrete enough to just start writing numbers down: $N = 11\ldots100\ldots0$, a ones then b zeros. A natural first move is to fix small values of a and b and test divisibility by $2020 = 2^2 \cdot 5 \cdot 101$ directly, hoping to spot when it happens – 110, 1100, 11100, and so on – checking each against 2020 by long division. This finds nothing quickly, because 101 is not a friendly modulus to eyeball: there is no small pattern in $\{1, 11, 111, \ldots\}$ mod 101 that jumps out from a handful of terms. The student can keep extending the search, but has no argument for when to stop, and no way to count N up to 2020 digits by trial. This is zombie maths: the right object (N as a function of two integer parameters a, b) has already been identified, but the next move is search instead of algebra.

The model chain. Write N in closed form: with $a \geq 1$ ones and $b \geq 0$ zeros,

$$N = \underbrace{11\cdots1}_{a}\underbrace{0\cdots0}_{b} = \frac{10^a - 1}{9} \cdot 10^b.$$

Condition (ii) is $a + b \leq 2020$. For condition (i), factor $2020 = 2^2 \cdot 5 \cdot 101$ and treat each prime power separately, because $\frac{10^a-1}{9}$ (a repunit) is always odd and 10^b carries all the even and 5-divisibility.

The $2^2 \cdot 5$ part. Since the repunit factor is odd, all of 2020's factor of 4 must come from 10^b, which needs $b \geq 2$; that single condition also supplies the factor of 5.

The 101 part. Since $\gcd(101, 10) = 1$, we need $101 \mid \frac{10^a-1}{9}$, and since $\gcd(9, 101) = 1$ this is equivalent to $101 \mid 10^a - 1$, i.e. $10^a \equiv 1 \pmod{101}$. Now $10^2 = 100 \equiv -1 \pmod{101}$, so $10^4 \equiv 1 \pmod{101}$ and the order of 10 modulo 101 is exactly 4 (it cannot be 1 or 2, since $10 \not\equiv 1$ and $10^2 \equiv -1 \not\equiv 1$). Hence $10^a \equiv 1 \pmod{101}$ iff $4 \mid a$.

So the count is over pairs (a, b) with $a \geq 1$, $4 \mid a$, $b \geq 2$, $a + b \leq 2020$. Write $a = 4k$, $k \geq 1$. For each such a, b ranges over $2, \ldots, 2020 - a$, giving $2019 - a$ choices whenever $a \leq 2019$, i.e. $a \leq 2016$ (the largest multiple of 4 not exceeding 2019), so $k = 1, \ldots, 504$. The count is

$$\sum_{k=1}^{504} (2019 - 4k) = 504 \cdot 2019 - 4 \cdot \frac{504 \cdot 505}{2} = 504 \cdot (2019 - 1010) = 504 \cdot 1009 = 508536.$$

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths agree that $N = \frac{10^a-1}{9} \cdot 10^b$ and that (ii) is $a + b \leq 2020$. They split at the question of *how* to test divisibility by 101 for a repunit. The natural attempt tests

numerically, one a at a time. The model asks instead: what is the order of 10 modulo 101? – and answers it in two lines using $10^2 \equiv -1$.

> *I don't want to test* $10^a \bmod 101$ *for each a one at a time – divisibility of* $10^a - 1$ *by a prime not dividing* 10 *is completely governed by the multiplicative order of* 10 *mod that prime, and I should compute that order once, by hand, from small powers, rather than search.*

> $100 \equiv -1 \pmod{101}$ *is not a coincidence to be looked up – it is the fact that* $101 = 100 + 1$*, so of course* $10^2 \equiv -1$*, and squaring that instantly gives the order:* 4*.*

Layer 3 THE WHY

The definition being used is the multiplicative order: for $\gcd(10, 101) = 1$, the set of a with $10^a \equiv 1 \pmod{101}$ is exactly the multiples of $\mathrm{ord}_{101}(10)$, because the powers of 10 modulo 101 cycle with period equal to that order. This is a structural fact about $(\mathbb{Z}/101\mathbb{Z})^\times$, not a property of the number 101 that has to be discovered by trial: any modulus coprime to 10 reduces the same way. Once that's internalised, computing $10^a - 1 \bmod 101$ for a specific a is never the right move; computing the order once is. And the order is cheap here specifically because $101 = 10^2 + 1$ is visibly one more than a perfect square, so $10^2 \equiv -1$ is read off the definition of 101 itself, not derived.

The decomposition of 2020 into coprime prime-power factors $4, 5, 101$ is likewise not a trick: divisibility by a product of pairwise coprime numbers is equivalent (by the Chinese Remainder Theorem) to divisibility by each factor separately, so splitting the problem into three independent, smaller conditions on (a, b) is simply unpacking what "divisible by 2020" means as efficiently as possible. Treating 2020 as one lump to be tested against would throw that structure away.

Caveat. I re-derived the entire count independently (order of 10 mod 101 is 4, condition $b \geq 2$, the arithmetic sum $504 \cdot 1009$) and it matches Kedlaya's stated answer of 508536 exactly, so this chain is fully verified, not merely paraphrased.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is the phrase "string of ones followed by zeros" translating directly into $N = \frac{10^a - 1}{9} \cdot 10^b$; the moment digits are described positionally like this, write down the closed form immediately rather than reasoning about digit strings.

2. That closed form suggests factoring the divisor 2020 into pairwise coprime prime powers and checking each one against the two "knobs" a (via the repunit) and b (via the power of 10) separately.

3. Hesitation occurs at the 101 condition, because $101 \nmid 10$ so no power of 10 trivially vanishes mod 101; the reinforcement is to recall that order-of-an-element arguments exist precisely for this situation and to look for a small relation like $10^2 \equiv \pm 1$ before resorting to a table.

4. The trap is running the same case-by-case numerical search on 101 that worked (barely) for spotting the $b \geq 2$ condition; 101 is not small enough for that to terminate usefully, and the problem is testing whether you switch tools.

5. Recognisable faster next time by treating "is $10^a \equiv 1$ mod a prime $p \nmid 10$" as an order computation on sight, and checking first whether p is one more or one less than a perfect square or has some other visible relation to a power of 10.

6. Reusable rule: when a divisibility condition involves $10^a - 1$ (or any $b^a - 1$) modulo a prime not dividing the base, compute the multiplicative order of the base modulo that prime directly – via a visible algebraic relation if one exists – rather than testing values of a numerically; the condition then collapses to a single congruence on a.

▷ **Cue:** a number described as a fixed string of digits at each position (e.g. a ones then b zeros) $\longrightarrow$ **Move:** write the closed form via $\frac{10^a-1}{9} \cdot 10^b$ immediately

▷ **Cue:** divisibility of $10^a - 1$ (or $b^a - 1$) by a prime not dividing the base $\longrightarrow$ **Move:** compute the multiplicative order of the base modulo that prime, not a per-value search

2020 A2

2020 A2. Let k be a nonnegative integer. Evaluate

$$\sum_{j=0}^{k} 2^{k-j} \binom{k+j}{j}.$$

Fields: combinatorics (binomial identities, probability) **Position:** A2 – second problem of the paper, an identity-evaluation whose clean closed form (4^k) is a strong hint that a structural argument exists **Answer:** 4^k.

Layer 1 THE WHAT — surface analysis

The natural attempt. The sum has an alternating structure in j and a binomial coefficient with a shifted top index, so the reflex is Pascal's rule: $\binom{k+j}{j} = \binom{k+j-1}{j} + \binom{k+j-1}{j-1}$, applied to peel the sum apart and relate $S_k = \sum_{j=0}^{k} 2^{k-j}\binom{k+j}{j}$ to S_{k-1}. Carrying this through (as the official first solution does) genuinely works: splitting the binomial, re-indexing the resulting sums, and matching terms against S_{k-1} and a piece of S_k itself eventually yields $S_k = 2S_{k-1} + S_k/2$, hence $S_k = 4S_{k-1}$, hence $S_k = 4^k$ from $S_0 = 1$. But the route there is several lines of index bookkeeping in which a sign or an off-by-one shift in the summation bounds is easy to introduce, and nothing about the derivation explains *why* the answer is a clean power of 4 until the very last line.

The model chain. Recognise the binomial coefficient $\binom{n-1}{k}2^{-n}$ as the probability, in an infinite sequence of fair coin flips, that the n-th flip is the $(k+1)$-th head overall (there must be exactly k heads among the first $n-1$ flips, probability $\binom{n-1}{k}2^{-(n-1)}$, times a further $1/2$ for flip n itself being heads). Let X be the (a.s. finite) trial number of the $(k+1)$-th head. Writing $n = k+j+1$ so that $j = 0, 1, 2, \ldots$ runs over all valid indices,

$$P[X = k+j+1] = \binom{k+j}{k}2^{-(k+j+1)} = \binom{k+j}{j}2^{-(k+j+1)}.$$

Summing over $j = 0, \ldots, k$ (i.e. $n = k+1, \ldots, 2k+1$) and multiplying by 2^{2k+1}:

$$2^{2k+1}\,P[X \le 2k+1] = \sum_{j=0}^{k} 2^{2k+1-(k+j+1)}\binom{k+j}{j} = \sum_{j=0}^{k} 2^{k-j}\binom{k+j}{j} = S_k.$$

So it remains only to find $P[X \le 2k+1]$, the probability that at least $k+1$ heads occur in the first $2k+1$ flips. Since $2k+1$ is odd, "at least $k+1$ heads" and "at most k heads" (equivalently at least $k+1$ tails) partition all outcomes into two events of equal size, by the symmetry that swaps every head with a tail: this bijection sends an outcome with $h \ge k+1$ heads to one with $2k+1-h \le k$ heads, so the two events have exactly the same count of outcomes and hence $P[X \le 2k+1] = \frac{1}{2}$. Therefore $S_k = 2^{2k+1} \cdot \frac{1}{2} = 2^{2k} = 4^k$.

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both chains treat $\binom{k+j}{j}2^{-j}$ as the object worth manipulating. They split at the very first decision about what to do with it: the natural attempt applies Pascal's identity to the binomial coefficient itself, staying inside pure algebra. The model chain instead asks what the coefficient *counts*, reads it as a probability, and reroutes the entire problem into a statement about coin flips before touching any algebra.

> *$\binom{n-1}{k}2^{-n}$ is not just an expression I can manipulate with Pascal's rule – it is literally the probability that a specific coin-flip event happens at trial n, so before I do any algebra I should ask what random experiment this sum is secretly counting.*

> *The bound $j \leq k$ is not an arbitrary cutoff to be pushed through re-indexing – in the coin-flip picture it is exactly the event "the $(k+1)$-th head arrives by trial $2k+1$", and $2k+1$ being odd is the whole reason the probability is forced to be exactly $1/2$ by a symmetry, rather than some messier fraction.*

Layer 3 THE WHY

The definition doing the work is the negative binomial probability mass function itself: $P[n\text{-th flip is the } (k+1)\text{-th head}] = \binom{n-1}{k}2^{-n}$ is not chosen for convenience, it is what "$(k+1)$-th head at trial n" *means* in terms of counting binary strings with a fixed number of heads in the first $n-1$ positions. Once the summand is seen as a probability, summing it over a range of j is automatically summing a probability over a range of n – no re-indexing trick is required, because that is simply what a partial sum of a probability mass function is.

The reason the final probability collapses to exactly $1/2$, with no further computation, is the symmetry of a fair coin under swapping heads and tails: this swap is a bijection on the 2^{2k+1} equally likely outcomes of $2k+1$ flips, and it exchanges "at least $k+1$ heads" with "at most k heads." Because $2k+1$ is odd there is no outcome with exactly $k+\frac{1}{2}$ heads sitting on the boundary to break the pairing, so the two events must have exactly equal probability, forcing each to be $1/2$. This is the deeper reason the sum evaluates to a power of 2 at all: the answer is not an accident of algebra, it is a symmetry of the fair coin made visible only once the sum is reinterpreted.

Caveat. I re-derived the coin-flip correspondence and the symmetry argument independently (rather than quoting Kedlaya's second solution) and checked it against $k = 0, 1, 2$ by direct computation of S_k; all match 4^k. The correspondence between $\binom{n-1}{k}2^{-n}$ and "n-th flip is the $(k+1)$-th head" is a standard fact about the negative binomial distribution and I have verified it from first principles (count binary strings of length n ending in a head with exactly k heads among the first $n-1$ positions).

Layer $+\alpha$ BACKTESTING

1. The first useful cue is a binomial coefficient of the exact shape $\binom{n-1}{k}$ multiplied by a power 2^{-n} (or, as here, an equivalent rearrangement of one): write down immediately that this is the probability mass function of "trial number of the $(k+1)$-th success" in fair Bernoulli trials.
2. That cue suggests recasting the whole sum as a probability (or expectation) statement about coin flips, then hunting for a symmetry of the coin-flip experiment that pins the probability down without computation, rather than manipulating the binomial coefficient directly.
3. Hesitation occurs at deciding what "event" the partial sum $j = 0$ to k corresponds to; the reinforcement is to always re-index by the natural variable of the combinatorial object (here $n = k + j + 1$, the trial number) rather than staying in j, since the natural variable is the one the probabilistic meaning is stated in.
4. The trap is trusting that Pascal's-rule algebra "always works eventually" and grinding through it, which it does here, but at the cost of several re-indexing steps and no insight into why the answer is exactly 4^k rather than some other power.
5. Recognisable faster next time by treating any sum of the form $\sum \binom{a+j}{j}p^{-j}$ or $\sum \binom{n-1}{k}p^n$ as a probability-generating-function pattern on sight, before reaching for Pascal's identity.

6. Reusable rule: when a sum's summand is literally a binomial probability (or can be rescaled into one), translate the whole sum into a probability of an event about repeated independent trials and look for a symmetry (parity of the trial count, exchangeability of outcomes) that evaluates that probability directly, instead of manipulating the binomial coefficients as pure algebra.

▷ **Cue:** a summand of the shape $\binom{n-1}{k}p^k(1-p)^{n-1-k}$ or a rescaled equivalent $\longrightarrow$ **Move:** reinterpret the whole sum as a probability statement about independent Bernoulli trials

▷ **Cue:** a partial sum truncated at an odd total that splits an experiment into two complementary events $\longrightarrow$ **Move:** look for a head/tail (or similar outcome-swapping) symmetry forcing the probability to be exactly $1/2$

2020 A3

2020 A3. Let $a_0 = \pi/2$, and let $a_n = \sin(a_{n-1})$ for $n \geq 1$. Determine whether

$$\sum_{n=1}^{\infty} a_n^2$$

converges.

Fields: analysis (recursive sequences, Taylor expansion, series convergence) **Position:** A3 – opens the analysis section of the paper; the recursion $a_n = \sin(a_{n-1})$ is a named classical object and the problem rewards recognising it **Answer:** The series diverges.

Layer 1 THE WHAT — surface analysis

The natural attempt. The instinct on seeing a series $\sum a_n^2$ built from an iterated sine is to try the standard comparison and ratio tests directly on a_n^2. Since $0 < a_1 < \pi/2$ and $\sin(x) < x$ for $x > 0$, the sequence a_n is positive, strictly decreasing, and $\to 0$, so $a_n^2 \to 0$ too – but that alone says nothing about convergence of the sum. A student then tries the ratio $a_{n+1}^2/a_n^2 = \sin(a_n)^2/a_n^2 \to 1$ as $n \to \infty$ (since $\sin(x)/x \to 1$), which is the borderline case the ratio test cannot resolve. Comparison to a geometric series fails for the same reason: a_n does not decay geometrically. The student is left knowing $a_n \to 0$ "slowly" but with no quantitative handle on *how* slowly, and no test in the standard toolkit that applies directly to a ratio limit of exactly 1.

The model chain. The ratio test failing at exactly 1 is the signal that the decay rate itself needs to be pinned down, so introduce $b_n = 1/a_n^2$ and expand sin to enough terms to see how b_n moves. Since $a_n \to 0$, for large n,

$$a_{n+1} = \sin(a_n) = a_n - \frac{a_n^3}{6} + O(a_n^5),$$

so

$$a_{n+1}^2 = a_n^2\left(1 - \frac{a_n^2}{6} + O(a_n^4)\right)^2 = a_n^2\left(1 - \frac{a_n^2}{3} + O(a_n^4)\right).$$

Taking reciprocals,

$$\frac{1}{a_{n+1}^2} = \frac{1}{a_n^2}\cdot\frac{1}{1 - \frac{a_n^2}{3} + O(a_n^4)} = \frac{1}{a_n^2}\left(1 + \frac{a_n^2}{3} + O(a_n^4)\right) = \frac{1}{a_n^2} + \frac{1}{3} + O(a_n^2).$$

That is, $b_{n+1} - b_n \to \frac{1}{3}$ as $n \to \infty$ (the $O(a_n^2)$ error term vanishes since $a_n \to 0$). By the Cesàro/Stolz averaging principle – if a sequence's consecutive differences tend to a limit L, the sequence itself grows like Ln – it follows that $b_n/n \to \frac{1}{3}$, i.e.

$$a_n^2 = \frac{1}{b_n} \sim \frac{3}{n} \qquad (n \to \infty).$$

Since $\sum 3/n$ diverges (the harmonic series), and a_n^2 is asymptotically a constant multiple of $1/n$, comparison gives that $\sum a_n^2$ diverges as well.

(A cruder but fully rigorous version of the same idea, sufficient by itself: since $\sin(x) \geq x - x^3/6$ for $x \in [0,1]$ by Taylor's theorem with remainder, one shows by induction that $a_n \geq 1/\sqrt{n}$ for all $n \geq 1$ – the inductive step reduces to the polynomial inequality $(n+1)(6n-1)^2 \geq 36n^3$, i.e. $(3n-1)(8n-1) \geq 0$, true for $n \geq 1$. Then $a_n^2 \geq 1/n$ termwise, and $\sum 1/n$ diverges, so $\sum a_n^2$ diverges by direct comparison, with no asymptotic or Stolz argument required.)

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both chains reach the same dead end: $a_n \to 0$ and $a_{n+1}^2/a_n^2 \to 1$, so the ratio test is silent. The natural attempt stops there, having exhausted the standard convergence tests. The model chain's next move is to stop asking "does this series converge" as a yes/no question about a_n^2 directly, and instead ask "at what rate does $a_n \to 0$" – converting a convergence question into an asymptotics question, one level of precision more demanding, and only then coming back to the series.

> *A ratio limit of exactly* 1 *is not a dead end, it's a message: the sequence decays at a rate the ratio test is too coarse to see, so I need to find that rate explicitly, and the standard way to find a decay rate for an iterated map near a fixed point is to Taylor-expand the map and look at the next-order term.*

> *I shouldn't work with a_n directly – I should invert it, because $a_{n+1} \approx a_n(1 - a_n^2/6)$ is a multiplicative statement close to* 1*, and multiplicative statements close to* 1 *turn into additive, trackable statements once you take reciprocals (or logs); $1/a_{n+1}^2 - 1/a_n^2 \to$ const is a statement I can sum.*

Layer 3 THE WHY

The reason the ratio test is silent is structural, not accidental: $a_n \to 0$ and $\sin(x)/x \to 1$ as $x \to 0$ together *force* $a_{n+1}/a_n \to 1$ for any sequence generated this way, so no fixed-point iteration of a smooth function with $f'(0) = 1$ (here sin) can ever be distinguished from a non-decaying sequence by the ratio test alone; the test is measuring the wrong order of the Taylor expansion. The information that actually governs the rate lives in the *second-order* term of $\sin(x) - x = -x^3/6 + O(x^5)$, which is why the correct move is to expand one order further than "$\sin(x) \approx x$" and track precisely the x^3 correction.

Passing to $b_n = 1/a_n^2$ is forced by the shape of that correction: because the leading discrepancy is cubic in a_n inside a squared quantity, $a_{n+1}^2 \approx a_n^2(1 - a_n^2/3)$ is a relation that is multiplicative in a_n^2 and additive once inverted, since $1/(1-\varepsilon) \approx 1 + \varepsilon$ turns a small multiplicative correction into a small additive one – this is exactly the mechanism (not a trick) by which telescoping differences of $1/a_n^2$ become tractable where differences of a_n^2 itself are not. This is a named, reusable pattern for any recursion $x_{n+1} = x_n - cx_n^{p+1} + O(x_n^{p+1+\delta})$: invert to the power p, and the resulting sequence has bounded, convergent *differences*, giving a clean polynomial asymptotic $x_n \sim (pc \cdot n)^{-1/p}$ via Stolz-Cesàro. Kedlaya's official solution instead proves only the weaker bound $a_n \geq 1/\sqrt{n}$ by induction on a Taylor inequality – sufficient for divergence, but it does not explain, as the asymptotic derivation does, exactly what constant a_n^2 is converging to relative to $1/n$ (namely 3, not merely "at least 1").

Caveat. The full asymptotic $a_n^2 \sim 3/n$ via the $b_n = 1/a_n^2$ telescoping argument is a standard technique for recursions of the form $x_{n+1} = \sin(x_n)$ (or more generally $x_{n+1} = x_n - cx_n^3 + \cdots$) and I have verified each algebraic step (the Taylor expansion, the reciprocal expansion, and the Stolz-Cesàro conclusion) myself; I have not additionally verified the fully rigorous $O(a_n^4)$ error bookkeeping needed to make "$b_{n+1} - b_n \to 1/3$" airtight rather than heuristic, and for that reason the model chain also gives the cruder, fully elementary

induction $a_n \geq 1/\sqrt{n}$ (matching Kedlaya's proof exactly) as the rigorously verified fallback establishing divergence on its own.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is a recursively defined sequence $a_{n+1} = f(a_n)$ where f fixes 0 and $f'(0) = 1$ (here $f = \sin$); on seeing this, write down immediately that the ratio test will be silent, because $f(x)/x \to 1$.

2. That cue suggests going one Taylor order deeper than the linear term to find the exact power-law decay rate, then converting the multiplicative recursion into an additive one via an appropriate power of a reciprocal.

3. Hesitation occurs right after confirming $a_{n+1}^2/a_n^2 \to 1$ and finding no applicable test; the reinforcement is to reframe the question from "does the series converge" to "how fast does $a_n \to 0$", since a convergence question about a sum is always secretly an asymptotics question about the terms.

4. The trap is concluding too early that $a_n \to 0$ "fast enough" from intuition (sine looks like it shrinks numbers quickly), when in fact the decay is only polynomial ($\sim n^{-1/2}$), not exponential, which is precisely what makes $\sum a_n^2$ diverge rather than converge.

5. Recognisable faster next time by memorising the pattern: $x_{n+1} = x_n - cx_n^{p+1} + \cdots \implies x_n \sim (pcn)^{-1/p}$, obtained by inverting to the p-th power and telescoping.

6. Reusable rule: for a recursion $x_{n+1} = f(x_n) \to 0$ with $f(x) = x - cx^{p+1} + O(x^{p+1+\delta})$ near 0, do not attempt the ratio or comparison test on x_n directly; instead set $b_n = 1/x_n^p$, show $b_{n+1} - b_n \to pc$ via the Taylor/reciprocal expansion, and conclude $x_n^p \sim 1/(pcn)$ by Stolz-Cesàro – this settles both the rate and any convergence question about $\sum x_n^q$ in one stroke.

▷ **Cue:** a recursion $a_{n+1} = f(a_n) \to 0$ with $f'(0) = 1$, so the ratio test gives limit 1 $\longrightarrow$ **Move:** expand f one Taylor order past linear to find the leading correction term

▷ **Cue:** a multiplicative near-1 correction $x_{n+1} \approx x_n(1 - cx_n^p)$ $\longrightarrow$ **Move:** invert to $b_n = 1/x_n^p$ and telescope the resulting additive differences via Stolz-Cesàro

2020 A4

2020 A4. Consider a horizontal strip of $N+2$ squares in which the first and the last square are black and the remaining N squares are all white. Choose a white square uniformly at random, choose one of its two neighbors with equal probability, and color this neighboring square black if it is not already black. Repeat this process until all the remaining white squares have only black neighbors. Let $w(N)$ be the expected number of white squares remaining. Find

$$\lim_{N\to\infty} \frac{w(N)}{N}.$$

Fields: probability: first-step analysis on a self-similar random process; combinatorics: derangements and the recurrence $D_n = (n-1)(D_{n-1} + D_{n-2})$ **Position:** A4 — the pivot of the paper. A1–A3 are single-idea problems; this is the first one where choosing the wrong state variable costs you the whole hour. **Answer:** $1/e$.

Layer 1 THE WHAT — surface analysis

The natural attempt. Number the squares 1 to $N+2$; squares 1 and $N+2$ are black. The first instinct is linearity of expectation, and it is a good instinct:

$$w(N) = \sum_{i=2}^{N+1} \mathbb{P}(\text{square } i \text{ is still white at the end}).$$

Now try to compute one of those probabilities. Square i survives if and only if, at every moment, nobody ever chose $i-1$ and pointed right, and nobody ever chose $i+1$ and pointed left. But whether $i-1$ is available to be chosen depends on whether $i-1$ is still white, which depends on what happened at $i-2$, which depends on $i-3$. The events do not decouple, the probability is not a function of i alone, and after twenty minutes you have a conditional probability with the entire history in the conditioning.

So you retreat to a Markov chain. What is the state? The obvious answer is "the number of white squares remaining", and this is the wall. Two configurations with the same number of whites behave completely differently: five whites in a row is a live system with four internal bonds, while five isolated whites separated by blacks is a terminal configuration with nothing left to do. The transition probabilities depend on the *arrangement*, not the count. Pushed honestly, the state space is the set of compositions of N into blocks of consecutive whites — exponentially many — and the chain is unmanageable.

A student who simulates instead will get $w(N)/N \approx 0.368$ and may even recognise $1/e$. That is worth something, and on an exam you should write the guess down. But the guess is not the problem, and simulation gives no route to the proof.

The model chain. Step 1: what a move actually is. A move chooses a white square i uniformly from the N whites and a direction, and blackens the neighbour if it is white. The moves that change nothing can be discarded: the final configuration is unaffected by them, and conditional on a change occurring, the ordered pairs (chosen white square, newly blackened white neighbour) are equally likely. There are exactly $2(N-1)$ such ordered pairs of adjacent white squares. Hence the first *newly* blackened square is square j with probability

$$\mathbb{P}(j) = \begin{cases} \frac{1}{2(N-1)}, & j = 2 \text{ or } j = N+1, \\ \frac{1}{N-1}, & 3 \le j \le N, \end{cases}$$

since square 2 can only be blackened from square 3, square $N+1$ only from square N, and an interior j from either side.

Step 2: the split. Once square j is black, the strip $1, \dots, N+2$ becomes two strips, $1, \dots, j$ and $j, \dots, N+2$, each with black ends and $j-2$ and $N+1-j$ whites respectively. No future move can act across j, because a move only ever blackens a neighbour of a white square and j is black. Moreover the process restricted to either piece is exactly the original process on that piece: a uniform choice among all remaining whites, conditioned to fall in one piece, is uniform among that piece's whites, and the final configuration does not depend on how the moves interleave between the pieces. So the two pieces evolve as independent copies of the problem, and

$$w(N) = \mathbb{E}\big[w(j-2) + w(N+1-j)\big].$$

Step 3: the recurrence. Writing that out and using $w(0) = 0$,

$$(N-1)\, w(N) = w(N-1) + 2\big(w(1) + \cdots + w(N-2)\big).$$

The right-hand side contains a sum of all earlier terms, so write the same identity with N replaced by $N-1$ and subtract:

$$(N-1)w(N) - (N-2)w(N-1) = w(N-1) - w(N-2) + 2w(N-2),$$

which rearranges to

$$\boxed{w(N) = w(N-1) + \frac{w(N-2)}{N-1}} \qquad (N \ge 2),$$

with $w(0) = 0$, $w(1) = 1$.

Step 4: identify the sequence. Compute: $w(0), \ldots, w(6) = 0,\ 1,\ 1,\ \frac{3}{2},\ \frac{11}{6},\ \frac{53}{24},\ \frac{103}{40}$. The denominators are factorials, so clear them: $N!\,w(N) = 0,\ 1,\ 2,\ 9,\ 44,\ 265,\ \ldots$. Those are the derangement numbers D_{N+1}. So the claim is

$$w(N) = \frac{D_{N+1}}{N!}, \qquad D_n = n! \sum_{k=0}^{n} \frac{(-1)^k}{k!}.$$

This is now an identity between two sequences with the same initial values, so it suffices to check the recurrences agree. Substituting,

$$w(N-1) + \frac{w(N-2)}{N-1} = \frac{D_N}{(N-1)!} + \frac{D_{N-1}}{(N-1)\,(N-2)!} = \frac{D_N + D_{N-1}}{(N-1)!},$$

and this equals $w(N) = D_{N+1}/N!$ precisely when $D_{N+1} = N\,(D_N + D_{N-1})$ — which is the standard derangement recurrence $D_n = (n-1)(D_{n-1} + D_{n-2})$. Done. Equivalently, in the form the official solution states,

$$w(N) = (N+1) \sum_{k=0}^{N+1} \frac{(-1)^k}{k!}.$$

Step 5: the limit. $w(N)/(N+1) = D_{N+1}/(N+1)!$ is exactly the probability that a uniformly random permutation of $N+1$ objects has no fixed point, and by the alternating series bound it differs from e^{-1} by at most $1/(N+2)!$. Since $(N+1)/N \to 1$,

$$\lim_{N\to\infty} \frac{w(N)}{N} = \frac{1}{e}.$$

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths agree on the opening move: linearity of expectation, and then "I need to understand the state of this process". At (†) — the choice of state variable — they split. The natural attempt takes the state to be *the number of white squares*, a single integer, and dies because the transition law depends on the arrangement. The model takes the state to be *the strip itself*, notices that one move cuts it into two strips of the same kind, and conditions on the cut. The natural attempt is asking "how does the count evolve?". The model is asking "what does one move do to the *shape* of the problem?"

> *The first square that actually turns black cuts the strip in two, and each half is the same problem again — black at both ends, white inside. So I should not be tracking a number through time. I should condition on where the single cut lands and let recursion do the rest.*

> *Most moves do nothing: I pick a white square whose chosen neighbour is already black. Those moves cannot affect the final picture, so I should throw them away and re-normalise. Conditioned on something happening, I am choosing uniformly among ordered adjacent white pairs, and I can count those:* $2(N-1)$.

> *My identity has* $w(1) + \cdots + w(N-2)$ *sitting in it — a running total of everything so far. Whenever I see that, I should write the identity one index down and subtract, because the difference of two running totals is a single term.*

> 0, 1, 1, $\frac{3}{2}$, $\frac{11}{6}$, $\frac{53}{24}$ — *those denominators are* $1, 1, 2, 6, 24$*, so the natural object is* $N!\,w(N)$*, not* $w(N)$*. Multiply through and look at the integers.*

Layer 3 THE WHY

Why must the cutting thought be thought? Because of what the stopping rule says. The process stops "when all the remaining white squares have only black neighbors". Read that as a structural statement rather than a termination condition: the terminal configuration is a set of isolated whites, each walled off by blacks. Walls are what this process manufactures. And a wall, the instant it appears, is absolute — no move ever crosses a black square, because a move only ever blackens a square *adjacent to a chosen white one* and it cannot un-blacken anything. So the process is not a random walk on configurations; it is a random *fragmentation*. The very first wall already partitions the problem permanently. Once you have said the word "wall", conditioning on the first one is not a trick, it is the only thing there is to do.

That also explains why the state "number of whites" had to fail. The count is not preserved by fragmentation in any useful way: two strips of sizes a and b are not equivalent to one strip of size $a+b$, because the strip of size $a+b$ has one more internal bond than the two pieces together. Bonds, not squares, are the currency — which is exactly why $2(N-1)$ and not $2N$ appears in Step 1, and why the factor $N-1$ rather than N appears in the recurrence.

Now the deeper why: where does $1/e$ come from, and is the appearance of derangements a coincidence? It is not. Look at the recurrence in its cleanest form, $D_n = (n-1)\big(D_{n-1} + D_{n-2}\big)$. The standard combinatorial reading of that recurrence is: in a derangement of $\{1, \ldots, n\}$, decide where element n goes — $n-1$ choices, say position j — and then split into two cases, according to whether j maps back to n (leaving a derangement of the remaining $n-2$ elements) or not (leaving a derangement of $n-1$ elements). One free choice of a location, then a two-way split into strictly smaller instances of the same problem. That is, structurally, the identical sentence to "choose where the first wall lands — roughly $N-1$ places — then split into two strictly smaller instances of the same problem". Kedlaya's solution asserts the closed form $w(N) = (N+1)\sum_{k \le N+1}(-1)^k/k!$ and verifies it by induction; here is where it comes from. It comes from the fact that this process and the fixed-point-free-permutation count are two readings of the same recursion, and e^{-1} is the derangement density.

> **Caveat.** The identity $w(N) = D_{N+1}/N!$ is proved above: both sides satisfy $w(N) = w(N-1) + w(N-2)/(N-1)$ with the same two initial values, and that determines the sequence. I have also checked it numerically against direct simulation of the process for $N \le 5$ and against exact rational computation of the recurrence for $N \le 200$. The *bijective* reading in the paragraph above — "choose a wall" matching "choose where n goes" — is offered as the explanation of why derangements appear, not as a proof; a genuine bijection is described in the official solution's remark (attributed to AoPS user pieater314159), and I have not verified that bijection line by line.

Finally, why $\lim w(N)/N$ rather than $\lim w(N)$? Because the answer is a *density*: in the limit, a fraction $1/e$ of the strip survives. That framing is the sanity check to run at the start. Each surviving white square needs both neighbours black, so survivors are isolated and the density is at most $1/2$; the process is greedy about making blacks, so the density should be comfortably below that; 0.368 is a plausible answer and 0.8 would not have been.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is the stopping rule: "until all the remaining white squares have only black neighbors". What should be written down on seeing it: *black squares are permanent walls, and this process is a fragmentation*. Immediately after that, write $w(0) = 0$, $w(1) = 1$, $w(2) = 1$ — the boundary data you will need.

2. The cue suggests first-step analysis with a *self-similar* split: condition on the first event, observe that it decomposes the instance into independent smaller instances of the same type, and write $w(N) = \mathbb{E}[w(\text{left}) + w(\text{right})]$.

3. Hesitation occurs at two places. First, at the choice of state: you feel that "number of whites" ought to work and spend time trying to patch it. What reinforces it away is the two-configuration test — write down five whites in a row and five isolated whites, both with count five, and see that one is terminal and one is not. Second, at the recurrence $(N-1)w(N) = w(N-1) + 2\sum_{j \le N-2} w(j)$, which looks intractable. What reinforces it away is the reflex: a running sum means shift-and-subtract.

4. Two traps. (i) Computing the probability of a move by dividing by $2N$ rather than $2(N-1)$ — that is, forgetting that moves which blacken an already-black square are null. Every constant in the answer depends on getting this right. (ii) Assuming the two halves are independent without saying why. They are, but the reason (moves never cross a black square, and the final configuration does not depend on the interleaving) is a real step and is exactly what a grader looks for.

5. Recognisable faster next time via the slogan: *a random process that creates permanent barriers is a fragmentation, and fragmentations are solved by conditioning on the first barrier.* Anywhere a process has irreversible moves that disconnect the state space — cutting a stick, splitting an interval, breaking a chain, the first record in a sequence — look for the self-similar split before looking for a Markov chain.

6. The reusable decision rule: **choose the state variable so that one step maps the problem into smaller copies of *itself*, not into a numerically simpler problem.** Corollaries. (a) If your candidate state is a count, test it on two configurations with the same count and different futures; if they differ, the count is not the state. (b) When a recurrence contains a running sum of all previous terms, write it one index down and subtract. (c) When the first few terms of a sequence have factorial denominators, multiply by $n!$ and read the integers — $1, 2, 9, 44, 265$ and $1, 1, 2, 6, 24$ and $1, 2, 5, 14, 42$ are worth knowing by sight.

▷ **Cue:** a random process whose moves are irreversible and create barriers that no later move can cross $\longrightarrow$ **Move:** condition on the first barrier; the two sides are independent instances of the same problem, so write $f(N) = \mathbb{E}[f(\text{left}) + f(\text{right})]$

▷ **Cue:** a move in a random process sometimes does nothing $\longrightarrow$ **Move:** discard the null moves and re-normalise; conditioned on a change, count the ordered pairs that produce a change and take them equally likely

▷ **Cue:** a recurrence of the form $c_N a_N = (\text{term}) + \sum_{j<N-1} a_j$ $\longrightarrow$ **Move:** write the identity with N replaced by $N-1$ and subtract, converting the running sum into one term

▷ **Cue:** the first terms of a rational sequence have denominators $1, 1, 2, 6, 24, 120$ $\longrightarrow$ **Move:** multiply by $n!$ and identify the integer sequence; derangements $1, 0, 1, 2, 9, 44, 265$ and their relatives appear constantly

▷ **Cue:** you are asked for $\lim a_N/N$ rather than $\lim a_N$ $\longrightarrow$ **Move:** the answer is a density; bound it crudely first (here: survivors are isolated, so the density is at most $1/2$) and use that to sanity-check the eventual constant

2020 A5

2020 A5. Let a_n be the number of sets S of positive integers for which

$$\sum_{k \in S} F_k = n,$$

where the Fibonacci sequence $(F_k)_{k \ge 1}$ satisfies $F_{k+2} = F_{k+1} + F_k$ and begins $F_1 = 1, F_2 = 1, F_3 = 2, F_4 = 3$. Find the largest integer n such that $a_n = 2020$.

Fields: combinatorics and number theory: Fibonacci representations, Zeckendorf's theorem, recursion on the top index, extremal bounds for a non-monotone counting function **Position:** A5 — hard. The counting function is easy to define, easy to compute, and wildly non-monotone, which is exactly what makes "largest n" the difficult half. **Answer:** $n = F_{4040} - 1$.

Layer 1 THE WHAT — surface analysis

The natural attempt. Compute. That is right, and you should. Note first the thing the statement is shouting: $F_1 = 1$ and $F_2 = 1$ are *different indices with the same value*, so $S = \{1\}$ and $S = \{2\}$ are different sets both summing to 1, and $a_1 = 2$. Then

$$a_1, \ldots, a_{10} = 2,\ 2,\ 3,\ 3,\ 3,\ 4,\ 3,\ 4,\ 5,\ 4.$$

Look at that sequence for thirty seconds and the difficulty of the problem becomes visible: $a_9 = 5$ but $a_{10} = 4$. The function is not monotone. It is not even eventually monotone. So "find the largest n with $a_n = 2020$" cannot be answered by finding where a_n crosses 2020, because it crosses repeatedly in both directions.

The natural attempt now goes hunting for the pattern in the record positions, and it finds one: the largest n with $a_n = 2$ is 2; with $a_n = 3$ is 7; with $a_n = 4$ is 20; with $a_n = 5$ is 54; with $a_n = 6$ is 143. Those are $F_4 - 1$, $F_6 - 1$, $F_8 - 1$, $F_{10} - 1$, $F_{12} - 1$. So the guess is $n = F_{4040} - 1$, and on an exam you write it down. Good.

The wall is the proof, and specifically the *minimality* half. You must show two things: that $a_{F_{4040}-1} = 2020$, and that $a_n \neq 2020$ for every one of the infinitely many $n > F_{4040} - 1$. The second is where the natural attempt dies, because the obvious strategy — prove a_n is increasing — is proving something false. A student who does not notice $a_9 > a_{10}$ will spend the rest of the hour trying to prove a false lemma. That is the most expensive thing that can happen on an A5.

The other natural attempt is combinatorial: start from the Zeckendorf representation of n (the unique way to write n as a sum of non-consecutive Fibonacci numbers) and count the other representations by repeatedly substituting $F_k = F_{k-1} + F_{k-2}$. This is genuinely productive — it is essentially the official second solution — and it gives good lower bounds on a_n. It is heavier to make exact.

The model chain. Throughout set $a_0 = 1$ (the empty set), and use the identity

$$F_1 + F_2 + \cdots + F_{m-1} = F_{m+1} - 1,$$

a one-line induction.

The key recursion. Fix $m > 0$ and let $F_m \leq n < F_{m+1}$. Any S with $\sum_{k \in S} F_k = n$ has all its indices at most m, since $F_{m+1} > n$. Split on whether $m \in S$.

- If $m \in S$, then $S \setminus \{m\}$ represents $n - F_m$. This is reversible: $n - F_m < F_{m+1} - F_m = F_{m-1} \leq F_m$, so *every* representation of $n - F_m$ automatically uses indices at most $m - 1$ and may be re-joined to $\{m\}$.
- If $m \notin S$, then $S \subseteq \{1, \ldots, m-1\}$, and the complement $T = \{1, \ldots, m-1\} \setminus S$ satisfies

$$\sum_{k \in T} F_k = (F_{m+1} - 1) - n.$$

  This is also reversible: $F_{m+1} - 1 - n \leq F_{m+1} - 1 - F_m = F_{m-1} - 1 < F_m$, so every representation of $F_{m+1} - 1 - n$ uses indices at most $m - 1$ and may be complemented back.

Hence

$$a_n = a_{\,n - F_m} + a_{\,F_{m+1} - 1 - n} \qquad (F_m \leq n < F_{m+1}). \tag{$\star$}$$

Note the two arguments sum to $F_{m-1} - 1$, so $(\star)$ is a statement about a single window folded onto itself.

Values at the window endpoints. Claim: for $m \geq 2$,

$$a_{F_m} = a_{F_{m+1}-1} = \left\lfloor \frac{m+2}{2} \right\rfloor.$$

Direct computation gives $a_1 = a_2 = 2$ and $a_3 = a_4 = 3$, covering $m = 2, 3, 4$. For $m \geq 5$, apply $(\star)$ at $n = F_m$: the two arguments are 0 and $F_{m+1} - 1 - F_m = F_{m-1} - 1$, so $a_{F_m} = a_0 + a_{F_{m-1}-1} = 1 + \lfloor m/2 \rfloor = \lfloor (m+2)/2 \rfloor$ by induction. Apply $(\star)$ at $n = F_{m+1} - 1$ (which lies in the window of m): the arguments are $F_{m+1} - 1 - F_m = F_{m-1} - 1$ and 0, so $a_{F_{m+1}-1} = a_{F_{m-1}-1} + a_0 = a_{F_m}$.

In particular, with $m = 4039$, $a_{F_{4040}-1} = \lfloor 4041/2 \rfloor = 2020$. So $n = F_{4040} - 1$ is achieved.

The floor on each window. Claim: for $F_m \leq n < F_{m+1}$, $a_n \geq a_{F_m}$. Induct on m. By the endpoint computation we may assume $F_m < n < F_{m+1} - 1$, so that in $(\star)$ both arguments $n - F_m$ and $F_{m+1} - 1 - n$ are at least 1; recall they sum to $F_{m-1} - 1$. Also assume $n \geq 6$, hence $m \geq 5$.

If $\max\{n - F_m,\ F_{m+1} - 1 - n\} \geq F_{m-2}$, then that summand contributes at least $a_{F_{m-2}} = \lfloor m/2 \rfloor$ by the inductive hypothesis, and the other contributes at least 2 (any a_j with $j \geq 1$ is at least 2, since $\{1\}$ and $\{2\}$ both represent 1 and can be adjoined). So $a_n \geq \lfloor m/2 \rfloor + 2 = \lfloor (m+4)/2 \rfloor > a_{F_m}$.

Otherwise both arguments are less than F_{m-2}, and since they sum to $F_{m-1} - 1 = F_{m-2} + F_{m-3} - 1$, both are at least F_{m-3}. Hence by induction

$$a_n \geq 2\, a_{F_{m-3}} = 2 \left\lfloor \frac{m-1}{2} \right\rfloor \geq m - 2 \geq \left\lfloor \frac{m+2}{2} \right\rfloor$$

for $m \geq 6$; the case $m = 5$ is checked directly against the computed table. This proves the claim.

Conclusion. If $n > F_{4040} - 1$ then $n \geq F_{4040}$, so n lies in the window of some $m \geq 4040$, and

$$a_n \;\geq\; a_{F_m} \;=\; \left\lfloor \frac{m+2}{2} \right\rfloor \;\geq\; \left\lfloor \frac{4042}{2} \right\rfloor = 2021 > 2020.$$

So no n beyond $F_{4040} - 1$ has $a_n = 2020$, and the answer is $\boxed{n = F_{4040} - 1}$.

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths agree through the data: compute a_n, spot the record positions $F_{2m} - 1$, write the answer down. They split at $(\dagger)$ — "now prove no larger n works". The natural attempt tries to prove a_n is increasing, or increasing from some point on. The model proves something strictly weaker and actually true: *a_n has a floor on each Fibonacci window, and that floor rises*. The difference is the difference between a false lemma and a true one, and it is decided by whether you looked at $a_9 = 5$, $a_{10} = 4$ hard enough to believe it.

> *$a_9 = 5$ and $a_{10} = 4$. So this function is not monotone and never will be, and any argument of the form "a_n passes* 2020 *and stays above it" is dead. What I actually need is a* lower bound valid for every n in a range*, with the range pushing rightwards. The natural ranges here are the Fibonacci windows $[F_m, F_{m+1})$, because those are the only intervals this problem knows about.*

> *If $F_m \leq n < F_{m+1}$, then no representation of n can use index $m+1$ or higher — it is too big. So the top index is forced to be at most m, and the only decision left at the top is: is m in S or not? Two branches, both of which must land back in the same counting problem, or the recursion is worthless.*

> *The branch "$m \notin S$" looks useless because it says S is an arbitrary subset of $\{1, \ldots, m-1\}$. But an arbitrary subset of a* fixed finite set *is the same information as its complement, and the complement's sum is (total) $- n$. The total $F_1 + \cdots + F_{m-1}$ is a Fibonacci number minus one. That*

> *is where $F_{m+1} - 1 - n$ has to come from; it is not invented.*

> *Look at where the recursion is cheapest. $a_n = a_{n-F_m} + a_{F_{m+1}-1-n}$, and the two arguments add to a constant. A sum of two increasing-ish things with a fixed total is smallest at the ends. At $n = F_m$ one argument is 0, contributing only $a_0 = 1$; anywhere strictly inside, both contribute at least 2. So the left endpoint of the window* must *be the minimum. I should prove that, not monotonicity.*

> *$F_1 = F_2 = 1$ is not a redundancy in the problem statement, it is the engine. It forces $a_j \geq 2$ for every $j \geq 1$, and that "≥ 2" is the quantity I multiply and add throughout the lower bound.*

Layer 3 THE WHY

Why must the window thought be thought? Because of what a Fibonacci representation *is*. Zeckendorf's theorem says every positive integer has a unique representation as a sum of non-consecutive Fibonacci numbers, obtained greedily; that representation is the Fibonacci numeral system, and in any numeral system the first question about n is "how many digits does it have?". Here "the number of digits" is precisely the index m with $F_m \leq n < F_{m+1}$. So the windows are not a clever choice of interval; they are the digits of n. Once you accept that, $(\star)$ is the ordinary digit recursion: peel off the leading digit, and recurse on what is left.

Why is the "$m \notin S$" branch a complementation? Because peeling off a leading digit of 0 leaves you with no information — the constraint has moved from "which indices are used" to "which indices are *not* used", and in a bounded index set those are the same data. This is the same move as counting the subsets of $\{1, \ldots, N\}$ with large sum by counting those with small sum. The reason $F_{m+1} - 1$ appears rather than some messier total is the identity $F_1 + \cdots + F_{m-1} = F_{m+1} - 1$, which is itself just the telescoping $F_k = F_{k+2} - F_{k+1}$. Nothing here is a rabbit.

Now the part the official solution states but does not motivate: why is the answer indexed at $4040 = 2 \times 2020$, and why floors of halves? Because the endpoint recursion is

$$a_{F_m} = 1 + a_{F_{m-1}-1}, \qquad a_{F_{m+1}-1} = a_{F_m},$$

so the value is *constant across the pair* $\{F_m,\ F_{m+1} - 1\}$ and goes up by exactly one when m increases by two. A quantity that gains one unit per two windows has index twice its value. The factor of 2 in F_{4040} is not arithmetic coincidence; it is the period of the recursion. The general lesson — and this is the transferable content — is: *when a recursion pairs up consecutive parameter values and increments once per pair, the answer's index is twice the target.* You will meet this again in problems about $\lfloor m/2 \rfloor$-type growth in tree recursions and in continued-fraction convergents.

And why does the left endpoint of each window minimise a_n? Drive it to the definition. $(\star)$ splits the representations of n into two families whose sizes are a_u and a_v with $u + v = F_{m-1} - 1$ fixed. The count a_j is at least 1 always and at least 2 as soon as $j \geq 1$, because $F_1 = F_2 = 1$ gives every positive j a free choice. So the only way to get a small total is to make one of u, v equal to 0, which happens exactly at $n = F_m$ and $n = F_{m+1} - 1$ — the two ends. Everywhere strictly inside, you pay at least $2 + 2$. That is the "of course": the endpoints are cheap because at an endpoint one of the two branches collapses to the empty representation. Kedlaya's Lemma 3 ("for $F_m \leq n < F_{m+1}$, $a_n \geq a_{F_m}$") is stated as a technical step; here is where it comes from, and here is why it is the right statement to attempt rather than monotonicity.

Caveat. I have verified the answer computationally. A subset-sum dynamic programme over the distinct indices $k \geq 1$ (with F_1 and F_2 both present as value 1) reproduces $a_1, \ldots, a_{10} = 2, 2, 3, 3, 3, 4, 3, 4, 5, 4$, and confirms that the largest n with $a_n = m$ is $F_{2m} - 1$ for $m = 2, \ldots, 8$ (namely $2, 7, 20, 54, 143, 376, 986$), and that $\min\{a_n : n \geq F_m\} = \lfloor (m+2)/2 \rfloor$

for $m \leq 15$. The proof above is complete except that the case $m = 5$ of the window-floor claim, and the small cases $n \leq 5$, are dispatched by the computed table rather than by a separate argument; both are finite checks and both hold.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is $F_1 = F_2 = 1$, spelled out in the statement when it did not have to be. Write down immediately: *indices, not values; so* $a_1 = 2$, *and* $a_j \geq 2$ *for all* $j \geq 1$. The second cue is the phrase "the largest integer n" — write down that this is an extremal claim about infinitely many n, so a lower bound on a whole tail is required.

2. The cue "sum of Fibonacci numbers with distinct indices" suggests the Fibonacci numeral system: locate n by its window $F_m \leq n < F_{m+1}$, peel off the top index, and recurse. That is Zeckendorf thinking, and it is the only standard technology that applies.

3. Hesitation occurs at the minimality half: you have the answer and no way to exclude larger n. What reinforces it away is computing ten more values of a_n and seeing the non-monotonicity explicitly — that kills the false lemma in ten seconds and redirects you to "bound a_n from below on each window". It also occurs at the "$m \notin S$" branch, which looks unstructured; what reinforces it away is the reflex to complement inside a bounded index set.

4. The trap is exactly the false monotonicity lemma. The problem is engineered for it: a_n rises steadily enough on the data a hurried student generates (a_1 through a_6 are $2, 2, 3, 3, 3, 4$, non-decreasing) that the drop at a_{10} is easy to miss. A secondary trap is forgetting $F_1 = F_2 = 1$ and computing $a_1 = 1$, which shifts every subsequent value and gives the wrong index parity.

5. Recognisable faster next time via the slogan: *a counting function defined by representations in a numeral system is controlled window by window, not term by term.* And more sharply: when asked for the largest n with $f(n) = c$ and f is not monotone, do not try to fix the monotonicity — find intervals on which f has a floor, and show the floors climb past c.

6. The reusable decision rule: **to prove "n_0 is the largest solution" for a non-monotone f, partition the tail (n_0, ∞) into natural intervals and prove a lower bound on each; choose the intervals so that the recursion defining f is uniform on them.** Two corollaries used here. (a) When a recursion reads $f(n) = f(u) + f(v)$ with $u + v$ constant on an interval, the minimum sits at the endpoints, where one term collapses — so prove endpoint formulas first and interior bounds second. (b) When counting subsets of a *bounded* index set by their sum, always ask whether the complement is easier; the complement's sum is the total minus yours, and for Fibonacci indices the total is $F_{m+1} - 1$.

▷ **Cue:** a sum over a set S of positive integers of F_k, with F_1 and F_2 both listed as 1 $\longrightarrow$ **Move:** treat indices as distinct labels, not values; conclude $a_j \geq 2$ for every $j \geq 1$ and keep that "≥ 2" as the unit of every later bound

▷ **Cue:** a quantity is indexed by n and the available structure is Fibonacci (or binary, or factorial) representations $\longrightarrow$ **Move:** locate n in its window $F_m \leq n < F_{m+1}$ — that window index is the number of digits — and recurse by peeling the top digit

▷ **Cue:** a branch of a case split says "S is an arbitrary subset of $\{1, \ldots, m-1\}$ with prescribed sum" $\longrightarrow$ **Move:** complement it: the complement has sum (total $-$ prescribed), and for Fibonacci indices the total is $F_1 + \cdots + F_{m-1} = F_{m+1} - 1$

▷ **Cue:** you are asked for the largest n with $f(n) = c$ $\longrightarrow$ **Move:** check monotonicity on twenty computed values before assuming it; if f is not monotone, switch to proving a rising floor on a partition of the tail

▷ **Cue:** a recursion has the shape $f(n) = f(u) + f(v)$ with $u + v$ fixed across an interval $\longrightarrow$ **Move:** the minimum of f on that interval is at the endpoints, where one of u, v is zero; compute the endpoints exactly and bound the interior by "both terms are at least the minimum"

2020 A6

2020 A6. For a positive integer N, let f_N be the function defined by

$$f_N(x) = \sum_{n=0}^{N} \frac{N + 1/2 - n}{(N+1)(2n+1)} \sin((2n+1)x).$$

Determine the smallest constant M such that $f_N(x) \le M$ for all N and all real x.

Fields: analysis: Fourier series, summation kernels, Fejér/Cesàro means, the Gibbs phenomenon; trigonometric summation by geometric series **Position:** A6 — the hardest problem on the paper, and one where the correct numerical answer is reachable by an argument that is flatly wrong. Say so and then dismantle it. **Answer:** $M = \pi/4$. The value is a supremum and is never attained: $f_N(x) < \pi/4$ for every N and every real x.

Layer 1 THE WHAT — surface analysis

The natural attempt. The odd harmonics $\sin((2n+1)x)$ with coefficients roughly $1/(2n+1)$ are recognisable on sight. The series

$$\sum_{n=0}^{\infty} \frac{\sin((2n+1)x)}{2n+1}$$

is the Fourier series of the square wave: it equals $\pi/4$ for $x \in (0, \pi)$, $-\pi/4$ for $x \in (-\pi, 0)$, and 0 at $x = 0, \pi$. So the natural attempt writes: "f_N is a partial sum of the square wave series, the square wave has supremum $\pi/4$, therefore $M = \pi/4$", and moves on.

The answer is right. The argument is wrong, and it is wrong in a way this problem was built to punish. Consider the honest partial sums

$$S_N(x) = \sum_{n=0}^{N} \frac{\sin((2n+1)x)}{2n+1},$$

with no weight attached. These do *not* stay below $\pi/4$. They overshoot near $x = 0$, by a fixed proportion that does not decay: this is the Gibbs phenomenon, and

$$\sup_N \sup_x S_N(x) \;=\; \tfrac{1}{2}\int_0^{\pi} \frac{\sin t}{t}\, dt \;=\; \tfrac{1}{2}\,\mathrm{Si}(\pi) \;\approx\; 0.92597,$$

against $\pi/4 \approx 0.78540$. The overshoot is about 17.9% and it never goes away. So "partial sums of a bounded function are bounded by its supremum" is false, and if the problem had asked about S_N the answer would have been $\mathrm{Si}(\pi)/2$, not $\pi/4$.

Which means the entire content of A6 lives in the weight

$$\frac{N + 1/2 - n}{(N+1)(2n+1)},$$

and a solution that does not explain what that weight does has not engaged with the problem. The natural attempt's wall is precisely that it treated the weight as cosmetic.

The model chain. *Step 1: read the weight.* Divide through:

$$\frac{N + 1/2 - n}{(N+1)(2n+1)} = \frac{2N + 1 - 2n}{2(N+1)(2n+1)} = \frac{1}{2n+1}\left(1 - \frac{2n+1}{2N+2}\right).$$

So, writing $k = 2n+1$ for the harmonic and $b_k = 1/k$ for the square wave's k-th sine coefficient ($b_k = 0$ for even k),

$$f_N(x) = \sum_{k \text{ odd}, k \le 2N+1} \left(1 - \frac{k}{2N+2}\right) b_k \sin(kx) = \sum_{k=1}^{2N+1} \left(1 - \frac{k}{(2N+1)+1}\right) b_k \sin(kx),$$

the even terms being absent because $b_k = 0$ there. That is verbatim the definition of the **Cesàro (Fejér) mean of order** $2N+1$ of the square wave g: $f_N = \sigma_{2N+1}(g)$. The triangular damping factor $1 - k/(M+1)$ is not a perturbation of the partial sum; it is the average of the first $M+1$ partial sums.

Step 2: the consequence. Fejér's theorem: $\sigma_M(g) = g * K_M$ where

$$K_M(t) = \frac{1}{M+1}\left(\frac{\sin\frac{(M+1)t}{2}}{\sin\frac{t}{2}}\right)^2 \;\geq\; 0, \qquad \frac{1}{2\pi}\int_{-\pi}^{\pi} K_M = 1.$$

A convolution with a nonnegative kernel of total mass 1 is an average, and an average cannot exceed a supremum. Since $\sup g = \pi/4$,

$$f_N(x) = (g * K_{2N+1})(x) \leq \sup g = \frac{\pi}{4} \qquad \text{for all } N, x.$$

Further, g is continuous on $(0,\pi)$, so Fejér's theorem gives $\sigma_M(g)(x) \to g(x) = \pi/4$ there. Taking $x = \pi/2$ and $N \to \infty$ shows no constant smaller than $\pi/4$ can work. Hence $M = \pi/4$. This is the whole problem, once the weight has been read.

Step 3: the self-contained version, for a write-up that quotes nothing. If you do not wish to cite Fejér, the positivity of the kernel is available directly. For $\sin x > 0$, summing a geometric series,

$$\sum_{n=0}^{N} \sin((2n+1)x) = \frac{1-\cos((2N+2)x)}{2\sin x} \;\geq\; 0. \tag{$\star$}$$

(Proof: write $\sin\theta = (e^{i\theta} - e^{-i\theta})/2i$, sum the two geometric series, and simplify; the numerator is $1-\cos$, hence nonnegative.) Now from the split $f_N = S_N - \frac{1}{2(N+1)}\sum_{n=0}^{N}\sin((2n+1)x)$ together with $(\star)$,

$$f_N(x) = \sum_{n=0}^{N} \frac{\sin((2n+1)x)}{2n+1} - \frac{1-\cos((2N+2)x)}{4(N+1)\sin x}. \tag{$\star\star$}$$

Comparing f_{N+1} with f_N using $(\star)$ — including or omitting the $n = N+1$ term according to the sign of $\sin((2N+3)x)$, so that the comparison is against a nonnegative partial sum either way — gives

$$f_{N+1}(x) - f_N(x) \;\geq\; \tfrac{1}{2}\left(\tfrac{1}{N+1} - \tfrac{1}{N+2}\right) \sum_{n=0}^{N \text{ or } N+1} \sin((2n+1)x) \;\geq\; 0$$

for $x \in (0,\pi)$. So $\{f_N(x)\}_N$ is nondecreasing on $(0,\pi)$. The correction term in $(\star\star)$ is $O(1/N)$ and vanishes, so $f_N(x) \to \sum_{n\geq 0}\sin((2n+1)x)/(2n+1) = \pi/4$ on $(0,\pi)$. A nondecreasing sequence with limit $\pi/4$ satisfies $f_N(x) \leq \pi/4$, with equality never attained.

Step 4: the rest of the real line. f_N is 2π-periodic and odd, so it suffices to bound it on $(-\pi,\pi]$. At $x = 0$ and $x = \pi$ every term vanishes, so $f_N = 0$. On $(-\pi,0)$, oddness turns the above into $0 \geq f_0(x) \geq f_1(x) \geq \cdots$, all negative. On $(0,\pi)$ we have $0 \leq f_0 \leq f_1 \leq \cdots < \pi/4$ with supremum $\pi/4$. Hence $M = \pi/4$ and nothing smaller works.

Where the maximum sits. Differentiating $(\star\star)$ and using a second geometric sum,

$$f_N'(x) = \frac{\cos x\,\big(1-\cos((2N+2)x)\big)}{4(N+1)\sin^2 x},$$

which is ≥ 0 on $(0,\pi/2]$ and ≤ 0 on $[\pi/2,\pi)$. So for every N the maximum over x is at $x = \pi/2$, where $\sin((2n+1)\pi/2) = (-1)^n$ and

$$f_N(\pi/2) = \sum_{n=0}^{N}\left(1 - \frac{2n+1}{2N+2}\right)\frac{(-1)^n}{2n+1} \;\longrightarrow\; 1 - \tfrac{1}{3} + \tfrac{1}{5} - \tfrac{1}{7} + \cdots = \arctan 1 = \frac{\pi}{4},$$

the Madhava–Gregory–Leibniz series. That reduces the whole problem to one classical evaluation.

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths agree on the first recognition: the harmonics $\sin((2n+1)x)$ with coefficients $\approx 1/(2n+1)$ are the square wave. They split at (†) — what to do with the factor $(N+1/2-n)/(N+1)$. The natural attempt treats it as a nuisance constant and discards it, reasoning about the Fourier series instead of about f_N. The model *divides it out and reads it*, getting $1-(2n+1)/(2N+2)$, and recognises a named object: the triangular damping of a Cesàro mean. Everything downstream is a consequence of that one act of reading.

> *That coefficient has N in it, and N is the summation limit. A coefficient that depends on where the sum stops is not a coefficient of a series — it is a* summation method. *So I should stop asking "what series is this?" and start asking "what averaging of the partial sums is this?"*

> *Divide the coefficient by the obvious* $1/(2n+1)$ *and see what is left:* $(N+1/2-n)/(N+1) = 1-(2n+1)/(2N+2)$. *A factor that decays linearly from* 1 *at the bottom harmonic to* 0 *at the top is the Fejér triangle. I have seen this exact shape before and it means: this is* σ_M, *not* S_M.

> *Before I claim* $\pi/4$, *let me check whether the* unweighted *partial sums stay below* $\pi/4$. *They do not — Gibbs overshoot, about* 18%, *and it does not decay. So the weight is doing all the work, and any argument of mine that would still run with the weight deleted is an argument that proves something false.*

> *A nonnegative kernel with total mass one turns convolution into a weighted average, and an average of a function never exceeds its supremum. That is the only reason a summation method can beat Gibbs, and it is the property I have to verify: is my kernel nonnegative?*

> *"Smallest constant M such that* $f_N(x) \le M$*" is a supremum over a two-parameter family, not a maximum. I should expect the bound to be approached and not attained, and I must supply both halves: an inequality valid for all* (N,x), *and a sequence of pairs* (N,x) *driving* $f_N(x)$ *up to it.*

Layer 3 THE WHY

Why must the weight be read rather than discarded? Because of the difference between a series and a summation method, and that difference is exactly what the Gibbs phenomenon exists to illustrate.

Drive it to the definitions. A Fourier partial sum is $S_M(g) = g * D_M$ where D_M is the Dirichlet kernel $\sin((M+\frac{1}{2})t)/\sin(\frac{t}{2})$. That kernel *changes sign* — it oscillates, and its L^1 norm grows like $\log M$. So $S_M(g)$ is not an average of g: it is a signed combination with total positive weight far exceeding 1, and there is no reason on earth for it to be bounded by $\sup g$. It is not. The overshoot near a jump is the kernel's negative lobes lining up with the jump, and $\mathrm{Si}(\pi)/2$ is the exact size of that misalignment.

Fejér's fix is not clever, it is the obvious repair: if a single partial sum overshoots because the kernel oscillates, average the partial sums so the oscillations cancel. Averaging $S_0,\dots,S_M$ replaces D by $K_M = \frac{1}{M+1}\sum_{j\le M} D_j$, and the algebra of that sum is a perfect square, $K_M(t) = \frac{1}{M+1}\big(\sin\frac{(M+1)t}{2}/\sin\frac{t}{2}\big)^2$. A perfect square is nonnegative. *That* is the whole of Fejér's theorem, and it is why a Cesàro mean obeys $\sigma_M(g) \le \sup g$ while a partial sum does not. And when you average the partial sums, the coefficient of $\sin(kx)$ gets multiplied by the fraction of the

partial sums that contain it, namely $(M+1-k)/(M+1) = 1-k/(M+1)$ — the triangle. *The triangular weight is not a weight; it is a bookkeeping record of how many partial sums each harmonic appeared in.* Once you know that sentence, the coefficient $(N+1/2-n)/((N+1)(2n+1))$ announces itself.

This also explains the elementary route, which is the same fact in disguise. The identity $(\star)$, $\sum_{n=0}^{N}\sin((2n+1)x) = (1-\cos((2N+2)x))/(2\sin x) \geq 0$, is a kernel positivity statement: the nonnegativity comes from $1-\cos \geq 0$, i.e. from a half-angle square, exactly as in K_M. Kedlaya's solution opens by rewriting f_N as $\frac{1}{2}\sum(\frac{2}{2n+1} - \frac{1}{N+1})\sin((2n+1)x)$ and then proving this sum is nonnegative; here is where that comes from. It is the conjugate Fejér kernel being nonnegative, proved by hand because the solution declines to name it.

Finally, why does the maximum sit at $x = \pi/2$ for every N, and why is that satisfying rather than lucky? Because g is the square wave: on $(0,\pi)$ it is constant, and its two jumps are at 0 and π. A nonnegative kernel concentrated at the origin, convolved with a function that is $+\pi/4$ on $(0,\pi)$ and $-\pi/4$ on $(-\pi,0)$, produces the largest value at the point furthest from both jumps — the point where the least mass of the kernel is sitting on the wrong side. That point is the midpoint, $x = \pi/2$. No calculus required to predict it; the derivative computation only confirms it.

Caveat. Everything above has been checked numerically. The weight identity $(N+1/2-n)/((N+1)(2n+1)) = (1-(2n+1)/(2N+2))/(2n+1)$ holds to machine precision for N up to 4000; $\sup_{x\in(0,\pi)} f_N$ equals $f_N(\pi/2)$ and rises through 0.5, 0.7626, 0.78292, 0.785148 for $N = 0, 10, 100, 1000$, against $\pi/4 = 0.7853982$, never reaching it; $f_N(x)$ is non-decreasing in N at sampled x; and $\sup_x S_N(x)$ for the unweighted partial sums sits at 0.9301, 0.92651, 0.925975 for $N = 3, 10, 100$, converging to $\mathrm{Si}(\pi)/2 = 0.9259682$ — confirming the Gibbs claim in the natural attempt. One correction to the source: the official solution's closing remark prints the derivative with a factor $\cos((N+2)x)$; the correct factor is $\cos((2N+2)x)$, as in the formula given above, which I have verified against numerical differentiation for several N and x. With $(N+2)x$ the stated formula is simply false.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is that the coefficient contains N, the upper limit of the sum. Write down immediately: *this is a summation method applied to a fixed series, not a series in its own right*. The second cue is $\sin((2n+1)x)/(2n+1)$: write down "square wave, value $\pi/4$ on $(0,\pi)$".

2. Those two cues together suggest: identify the underlying function (g = square wave), identify the method (divide out $1/(2n+1)$ and look at what remains — here $1-k/(M+1)$, the Fejér triangle), and then use the method's kernel to bound the result by $\sup g$.

3. Hesitation occurs at the moment you have the number $\pi/4$ and no proof — the temptation is to write "it is the Fourier series of the square wave" and stop. What reinforces it away is one concrete check: compute $\sup_x S_N(x)$ for $N = 10$ numerically, or just recall that Gibbs overshoot exists, and observe that it exceeds $\pi/4$. That single fact tells you the weight is load-bearing and forces you back to reading it. Hesitation occurs again at proving the sup is not smaller; what reinforces it away is remembering that "smallest M" always needs two halves, an inequality and an approaching sequence.

4. The trap is the right answer by a broken argument — rare, deliberate, and worth naming. The problem is engineered so that discarding the weight gives $\pi/4$ anyway, so a student can produce a confident wrong proof and never find out. The secondary trap is forgetting x ranges over *all* reals: you must say that f_N is odd and 2π-periodic and that the negative half is bounded above by 0, or your proof only covers $(0,\pi)$.

5. Recognisable faster next time via the slogan: *if the coefficients depend on where the sum stops, you are looking at a summation method, and the question is about the method's kernel, not the series.*

Learn the two kernels by their shapes: Dirichlet (oscillating, L^1 norm $\sim \log M$, overshoots) and Fejér (a perfect square, nonnegative, mass one, never overshoots). Triangular damping $1 - k/(M+1)$ means Fejér; a sharp cut-off means Dirichlet.

6. The reusable decision rule: **to bound something that is an average, exhibit the averaging measure and check it is nonnegative with total mass one; the bound is then free.** Everything of the form $\int f\,d\mu$ with $\mu \geq 0$ and $\mu(\text{all}) = 1$ satisfies $\inf f \leq \int f\,d\mu \leq \sup f$, and that single line replaces an optimisation. The converse discipline is equally reusable: *whenever a bound of that shape fails, look for a sign change in the weight*, because a signed weight of total mass one can overshoot without limit. This is the same principle behind Jensen's inequality, behind why positive quadrature rules are stable and oscillatory ones are not, and behind why smoothing a noisy signal with a nonnegative kernel cannot create new extrema.

▷ **Cue:** the coefficients of a finite trigonometric sum depend on N, the upper limit of summation $\longrightarrow$ **Move:** stop treating it as a series; divide out the natural coefficient and read what remains as a summation method, then work with that method's kernel

▷ **Cue:** a damping factor of the form $1 - k/(M+1)$ multiplies the k-th Fourier coefficient $\longrightarrow$ **Move:** this is the Fejér/Cesàro mean σ_M; the kernel is a perfect square, hence nonnegative with mass one, hence $\sigma_M(g) \leq \sup g$

▷ **Cue:** you must bound a quantity that is a convolution or a weighted average $\longrightarrow$ **Move:** show the weight is nonnegative with total mass one; then inf $\leq$ quantity $\leq$ sup with no further work

▷ **Cue:** a Fourier partial sum is claimed to be bounded by the supremum of its function $\longrightarrow$ **Move:** disbelieve it — the Dirichlet kernel changes sign and Gibbs overshoot is about 17.9%; the claim is true only for summation methods with nonnegative kernels

▷ **Cue:** you need $\sum_{n=0}^{N} \sin((2n+1)x)$ or a similar one-sided trigonometric sum $\longrightarrow$ **Move:** write $\sin\theta = (e^{i\theta} - e^{-i\theta})/2i$ and sum two geometric series; here the result is $(1 - \cos((2N+2)x))/(2\sin x)$, whose sign is free because $1 - \cos \geq 0$

▷ **Cue:** the question asks for the smallest constant M with $F \leq M$ $\longrightarrow$ **Move:** supply two halves — an inequality valid for the whole family, and an explicit sequence in the family approaching M — and state whether M is attained

Putnam 2020 — Paper B

2020 B1

2020 B1. For a positive integer n, define $d(n)$ to be the sum of the digits of n when written in binary (for example, $d(13) = 1+1+0+1 = 3$). Let

$$S = \sum_{k=1}^{2020} (-1)^{d(k)} k^3.$$

Determine S modulo 2020.

Fields: Number theory, generating functions **Position:** B1 – opener of the second half; looks like a grinding computation but is really a test of whether you reach for the generating-function identity behind binary digit sums **Answer:** $S \equiv 1990 \pmod{2020}$.

Layer 1 THE WHAT — surface analysis

The natural attempt. The instinct on seeing "sum weighted by $(-1)^{d(k)}$" is to pair consecutive integers. Since the binary expansion of $2j$ is the expansion of j with a 0 appended, $d(2j) = d(j)$; since $2j+1$ appends a 1, $d(2j+1) = d(j)+1$, so $(-1)^{d(2j+1)} = -(-1)^{d(2j)}$. Pairing $k = 2j$ with $k = 2j+1$ and expanding the cube,

$$(-1)^{d(2j)}(2j)^3 + (-1)^{d(2j+1)}(2j+1)^3 = (-1)^{d(j)}\left[(2j)^3 - (2j+1)^3\right] = -(-1)^{d(j)}(12j^2+6j+1).$$

This genuinely reduces the range in half, and it is a correct method – the trouble is what it reduces *to*. The cubic sum has now become a mixture of a quadratic sum $\sum(-1)^{d(j)}j^2$, a linear sum $\sum(-1)^{d(j)}j$, and a constant sum $\sum(-1)^{d(j)}$, each over a range half the size. Each of those needs its own version of the same pairing trick, and each spawns a lower-degree sum of its own. A student who starts this way is doing real, correct work, but they have turned one cubic sum into a system of four sums of decreasing degree, tracked over $\log_2(2020) \approx 11$ halvings, with boundary terms at every odd endpoint. It terminates, but only after a page of bookkeeping across four interleaved recursions – exactly the kind of real-but-untargeted grinding that eats a B1 budget.

The model chain. Work with the generating function. Multiplying out

$$(1-x)(1-x^2)(1-x^4)\cdots(1-x^{1024})$$

means choosing, from each factor $(1-x^{2^j})$ for $j = 0,\dots,10$, either the term 1 or the term $-x^{2^j}$. A choice of "$-x^{2^j}$" from a subset $T \subseteq \{0,\dots,10\}$ of the factors contributes the monomial $(-1)^{|T|}x^{\sum_{j\in T}2^j}$. Every integer k with $0 \le k \le 2047$ has a unique binary expansion, i.e., a unique subset T of $\{0,\dots,10\}$ with $k = \sum_{j\in T}2^j$, and $|T| = d(k)$. So the coefficient of x^k in the product is exactly $(-1)^{d(k)}$:

$$(1-x)(1-x^2)(1-x^4)\cdots(1-x^{1024}) = \sum_{k=0}^{2047}(-1)^{d(k)}x^k.$$

Applying $x\frac{d}{dx}$ three times turns the right side into $\sum_{k=0}^{2047}(-1)^{d(k)}k^3x^k$ (each application of $x\,d/dx$ multiplies the coefficient of x^k by k without disturbing the exponent). On the left, the product is a product of 11 factors, each of which has a *simple* zero at $x=1$ (since $\frac{d}{dx}(1-x^{2^j}) = -2^j\cdot 2^{j\cdot 0} \neq 0$ at $x=1$), so the whole product vanishes to order exactly 11 at $x=1$. Three derivatives cannot dislodge a zero of order 11, so the differentiated product still vanishes at $x=1$. Hence

$$\sum_{k=0}^{2047}(-1)^{d(k)}k^3 = 0.$$

The identical argument on $x^{2016}(1-x)(1-x^2)(1-x^4)(1-x^8)(1-x^{16}) = \sum_{k=2016}^{2047}(-1)^{d(k)}x^k$ – a product with only 5 vanishing factors, still more than the 3 derivatives taken – gives

$$\sum_{k=2016}^{2047}(-1)^{d(k)}k^3 = 0.$$

Subtracting, $\sum_{k=0}^{2015}(-1)^{d(k)}k^3 = 0$, and since the $k=0$ term is 0, $\sum_{k=1}^{2015}(-1)^{d(k)}k^3 = 0$. Therefore

$$S = \sum_{k=2016}^{2020}(-1)^{d(k)}k^3.$$

Working modulo 2020, $2016\equiv -4$, $2017\equiv -3$, $2018\equiv -2$, $2019\equiv -1$, $2020\equiv 0$. Computing $d(2016)=6$, $d(2017)=7$, $d(2018)=7$, $d(2019)=8$, $d(2020)=7$ (binary $2016 = 11111100000_2$ and its four successors), the signs are $+,-,-,+,-$, so

$$S \equiv (-4)^3-(-3)^3-(-2)^3+(-1)^3-0 = -64+27+8-1 = -30 \equiv 1990 \pmod{2020}.$$

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths start from the same observation: $d(2j)=d(j)$, $d(2j+1)=d(j)+1$. The natural attempt uses this fact to recurse on the *arithmetic of* k – split even from odd, again and again, and watch the exponent 3 spawn exponents $2,1,0$ along the way. The model uses the same fact to recurse on the *structure of the generating function* – recognizing that $d(k)$ is literally counting which factors of $(1-x)(1-x^2)(1-x^4)\cdots$ contributed a minus sign, so the whole weighted sum, for *any* fixed power of k, is a single coefficient-extraction problem, not a family of problems indexed by degree.

> *The exponent* 3 *is not the obstacle here – it's a distraction. What I actually need is a run of* 2^m *consecutive integers starting at a multiple of* 2^m*, because a run like that is forced to sum to zero against* any *polynomial weight up to degree* $m-1$*, by a fact about the order of vanishing of a product at* $x=1$*, not by anything special about cubes.*

> *To land the leftover piece near* 2020*, I should work backwards from the target:* 2016 *is the largest multiple of* $32=2^5$ *at or below* 2020*, so if I isolate the block* $[2016, 2047]$ *I get a run of exactly* 32 *integers whose weighted sum vanishes using only* 5 *dyadic factors – comfortably more than the* 3 *derivatives I need – leaving a residual of just* 5 *numbers,* 2016 *through* 2020*, to add up by hand.*

Layer 3 THE WHY

The recursive rule $d(2j)=d(j)$, $d(2j+1)=d(j)+1$ is not an isolated cute fact about binary digits – it is the precise statement that $(-1)^{d(k)}$ is the coefficient function of the product $\prod_j(1-x^{2^j})$, because "the binary expansion of k" and "the subset of factors from which $-x^{2^j}$ was chosen" are the same object. Once a sum is recognised as a coefficient in a known product, questions

about power-weighted sums of those coefficients (here, $\sum(-1)^{d(k)}k^3$) become questions about the derivatives of that product at $x = 1$, and derivatives of a product of simple factors at a common zero are controlled entirely by counting: a product of $m+1$ factors, each vanishing to order 1 at $x = 1$, vanishes to order exactly $m+1$ there, and no derivative of order less than $m+1$ can see past that. This is why the archive keeps returning to "identify the generating function first": once you have it, questions about specific numerical power sums stop being separate problems and become one fact (order of vanishing) applied twice, with different numbers of factors.

The second half of the argument – why 2016 to 2020 and not some other five numbers – follows from the same idea used defensively. You want the two blocks you subtract, $[0, 2047]$ and $[2016, 2047]$, to both be provably zero using *fewer* dyadic factors than the number of derivatives you're taking (here, 3), and you want their difference, $[0, 2015]$, to be exactly the numbers below the messy target range $[1, 2020]$. Choosing $2016 = 2^5 \cdot 63$ as the split point is forced once you ask "what is the largest power-of-two multiple at or below 2020" – it is not a guess, it is the natural stopping point of the same bit-counting idea applied to the boundary itself.

Caveat. None required for this problem: the vanishing-order argument, the digit computations for 2016–2020, and the final modular arithmetic were all independently re-derived and checked against the official answer, which they match exactly.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is the phrase "sum of the digits in binary" appearing inside a signed sum – on seeing it, write down the doubling recursion $d(2j) = d(j)$, $d(2j+1) = d(j)+1$ immediately, before deciding how to use it.

2. That recursion suggests treating $(-1)^{d(k)}$ as the coefficient sequence of the product $\prod_j(1-x^{2^j})$, and extracting power-weighted sums via $(x\,d/dx)^r$ evaluated at $x = 1$.

3. Hesitation occurs right after the natural pairing trick appears to work on the cubic term – it is tempting to keep pairing, not noticing that each pairing step manufactures a new, lower-degree sum that needs its own treatment. The reinforcement is to ask, before continuing, "how many distinct power sums am I now tracking?" – if the answer is growing, you're solving the wrong-shaped problem.

4. The trap is treating the exponent 3 as central and trying to find a closed form for cubic power sums weighted by $(-1)^{d(k)}$ directly; no such elementary closed form exists in general, and only full dyadic blocks vanish.

5. Recognisable faster next time by naming the object: "binary digit sum inside a sign" is always $\prod(1-x^{2^j})$ in disguise, regardless of what power of k is being weighted.

6. Reusable rule: to evaluate $\sum_k(-1)^{d(k)}f(k)$ for a polynomial f over an awkward range, write the range as a difference of two blocks $[0, 2^m-1]$ that are both provably zero (choose m large enough that the number of dyadic factors exceeds $\deg f$), leaving only a short residual block to sum by hand.

▷ **Cue:** $(-1)^{d(k)}$ inside a sum over consecutive integers, weighted by a polynomial in k $\longrightarrow$ **Move:** write it as the coefficient sequence of $(1-x)(1-x^2)(1-x^4)\cdots$ and extract the power sum via $(x\,d/dx)^r$ at $x = 1$, using the order of vanishing at $x = 1$ to force it to zero over full dyadic blocks

▷ **Cue:** need to isolate a short numeric range inside a sum that is zero over full dyadic blocks $\longrightarrow$ **Move:** write the target range as a difference of two blocks, each starting at 0 or a multiple of a power of two, so both blocks vanish and only a short residual remains

2020 B2

2020 B2. Let k and n be integers with $1 \leq k < n$. Alice and Bob play a game with k pegs in a line of n holes. At the beginning of the game, the pegs occupy the k leftmost holes. A legal move consists of moving a single peg to any vacant hole that is further to the right. The players alternate moves, with Alice playing first. The game ends when the pegs are in the k rightmost holes, so whoever is next to play cannot move and therefore loses. For what values of n and k does Alice have a winning strategy?

Fields: Combinatorial game theory **Position:** B2 – the second problem of the second half, an impartial finite game whose answer is a parity condition, disguised behind a counting-heavy setup **Answer:** Alice has a winning strategy if and only if at least one of n and k is odd (equivalently, Bob wins exactly when n and k are both even).

Layer 1 THE WHAT — surface analysis

The natural attempt. The game is finite, impartial (both players face the same set of legal moves from a given position), and terminates, so the first instinct – a correct one – is Sprague–Grundy: every position is a loss for the player to move (a P-position) or a win (an N-position), and you can in principle compute this by working backwards from the terminal position. A student sets up small cases by hand: $n = 2, k = 1$; $n = 3, k = 1$; $n = 3, k = 2$; $n = 4, k = 1$; $n = 4, k = 2$; and so on, tabulating who wins. This works, and after enough small cases a pattern does emerge – Alice wins unless n, k are both even. But this approach has produced a *conjecture*, not a proof. To turn "the pattern holds for every case I computed" into "the pattern holds for all n, k" by this route, you would need an inductive argument over the full Grundy-value recursion, tracking how the value of an (n, k)-position depends on the values of all positions reachable by a single move – a combinatorial explosion of states as soon as k is more than 2 or 3, since a position is a k-subset of n holes, not a single number. Brute-force case analysis finds the answer; it does not supply a proof that fits in a Putnam-length argument.

The model chain. Reduce every case to a single base case, then prove that base case with an explicit strategy rather than a computation. First handle n, k both even directly: partition the n holes into $n/2$ pairs of adjacent holes, $\{1, 2\}, \{3, 4\}, \ldots, \{n-1, n\}$, and call a configuration *good* if, for every pair, either both holes are occupied or neither is. The starting position – pegs in the k leftmost holes, with k even – is good, since it fills whole pairs. Claim: Bob can always return the position to a good one. From a good position it is Alice's turn; any move she makes takes a peg out of some occupied pair (breaking it to one-occupied) and puts it into some hole of a not-yet-full pair (also leaving that pair with one occupied hole, unless it was fully vacant and stays with exactly one occupied). Bob then moves the *other* peg of the pair Alice just broke into the remaining open hole of the pair Alice just touched, restoring a good configuration. This response is always available because Alice's move always leaves exactly one broken pair with a peg still in it (the one she moved from) and exactly one pair that just gained a single peg (the one she moved to), and Bob's move exists precisely because those two "half-full" pairs can be paired off. Since a good configuration always admits a legal response for Bob, and the game is finite, Bob never runs out of moves before Alice does, so Bob wins whenever n, k are both even.

Now handle the other three parities by reducing them to this base case with a single explicit first move by Alice, after which it is Bob who has just moved first into an even-even game with the roles swapped:

- n, k both odd: Alice moves the peg currently in hole k to hole n (the rightmost hole is vacant since $k < n$). This peg, and hole n, are now permanently out of play (no peg can ever be moved into a hole occupied by a peg that has already reached its final position beyond which nothing moves it again, and no earlier peg can pass it). What remains is exactly the $(n-1, k-1)$-game, both terms now even, with Bob to move first – so Bob is the "first player" of an even-even game, hence the losing player, so Alice (who moves second in the subgame) wins.

- n odd, k even: Alice moves the peg in hole 1 to hole n. Wait – more precisely, she moves the leftmost peg out to hole $k+1$, vacating hole 1 permanently (no peg can ever occupy hole 1 again, since pegs only move rightward and hole 1 was leftmost). This reduces to the $(n-1, k)$-game with Bob to move, again both even, so Alice wins by the same logic.
- n even, k odd: Alice moves the peg in hole 1 to hole n, removing both hole 1 and hole n (and this peg) from play at once. This reduces to the $(n-2, k-1)$-game with Bob to move, both even, so Alice wins.

In every case where at least one of n, k is odd, Alice has an explicit first move reducing to an even-even subgame with Bob forced to move first in it, and the base case shows the player forced to move first in an even-even game loses.

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths begin the same way: recognize the game is impartial and finite, so some fixed parity-based answer is plausible. They split at the question "how do I turn the pattern I've observed into a proof?" The natural attempt keeps computing – more small cases, more Grundy tables, hoping the pattern's proof falls out of enough examples. The model attempt stops computing states and instead asks: "is there a fixed structure (a matching of the holes) that one player can maintain as an invariant no matter what the other player does?"

> *I don't want to compute Grundy values position by position – I want a fixed partition of the n holes into pairs, defined once and for all before the game starts, such that the starting configuration respects it and every move by my opponent is forced to leave exactly one pair "half-broken," because then I have a guaranteed response that repairs it, and I never have to look at the game tree again.*

> *The three odd cases are not separate problems – each one is "one hole (or one hole plus its mirror, plus the peg that has reached it) is going to be permanently irrelevant to the rest of the game, so let me spend my very first move making that hole and that peg irrelevant on purpose," because doing so converts an odd case into the even case I already solved, with the extra bonus that my opponent is now the one forced to move first into it.*

Layer 3 THE WHY

The definition of the "good" invariant is not an arbitrary bookkeeping device – it exists only because n is even, which is exactly what lets the n holes be partitioned into $n/2$ disjoint adjacent pairs with nothing left over, and k is even, which is exactly what lets the starting configuration (pegs filling the k leftmost holes) respect that partition by filling whole pairs rather than half of one. If either n or k were odd, there would be no way to define "good" so that the starting position satisfies it – the invariant's existence is a direct consequence of the parity hypothesis, not a coincidence discovered after the fact. This is why the problem's answer is exactly "both even": that is precisely the condition under which the pairing structure the proof depends on can be built at all.

The reason Bob's response always exists is a one-move lookahead fact about the pairing itself: from a good position, any single legal move touches exactly two pairs – the one it vacates a hole from, and the one it fills a hole in – and leaves each with exactly one occupied hole, because a legal move only ever changes the occupancy of the hole moved from and the hole moved to. Bob's job is never to search for a good response among many options; there is exactly one hole left to move to restore goodness (the other hole of the pair Alice just partially filled), and exactly one peg available to move there (the one left behind in the pair Alice just partially emptied). The invariant does the entire strategic burden; Bob is executing an algorithm, not making a decision.

The reductions for the odd cases work because a peg that reaches the rightmost hole n can never move again (there is no hole further right), and a hole that has been permanently vacated at the far left can never be filled again (pegs only move rightward). So removing a peg-and-hole pair from either end of the board genuinely deletes them from the game rather than merely relabeling it – the remaining pegs and holes form a completely independent instance of the same game, smaller by exactly the amount removed, with no interaction possible between the frozen piece and the rest.

Caveat. I have directly verified the logic of the pairing invariant (that every legal move from a good position leaves exactly one hole available for Bob's repair move) and the logic of all three odd-case reductions (that a peg reaching hole n, or a hole 1 being vacated, is permanently removed from play). I have not independently brute-force-verified the Grundy tables for a range of small (n, k) against this claimed criterion by computer; the small hand-checked cases ($n = 2, k = 1$ and the structure of the $n = 4, k = 2$ pairing) are consistent with the claim, but a systematic computational cross-check was not performed here.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is that the game is impartial, finite, and terminates in a fixed final position regardless of play – on seeing this, write down "P-position/N-position, and check whether the answer is likely to be parity-based" before doing anything else.

2. That cue suggests looking for an invariant-based strategy (a pairing or matching that one player restores every turn) as a cheaper alternative to full Grundy-value computation.

3. Hesitation occurs right after guessing the parity pattern from small cases: it is tempting to keep computing more cases for confidence rather than asking what structural fact about the board would make "both even" special. The reinforcement is to ask directly, "what changes about the board's own structure when n and k are both even, that isn't true otherwise?" – here, that the n holes split evenly into pairs and the starting position respects that split.

4. The trap is assuming the invariant must be found by trial and error on the pairing shape; the correct pairing (adjacent holes $\{1, 2\}, \{3, 4\}, \dots$) is forced by wanting the starting position to already be good, not chosen for its elegance.

5. Recognisable faster next time by naming the mechanism: "parity-based winner in an impartial game" should trigger "look for a pairing invariant the target player restores every move," checked first against whether the pairing is consistent with the starting position.

6. Reusable rule: in an impartial combinatorial game where the winner is suspected (from small cases) to depend on a parity condition, first look for an explicit pairing/matching invariant that the target player can always restore after the opponent's move, and handle the "wrong parity" cases by finding a single explicit first move that fixes the parity and reduces to the base case with the opponent now forced to move first into it.

▷ **Cue:** an impartial, finite game whose small-case winner appears to depend on the parity of a count $\longrightarrow$ **Move:** search for a pairing/domino invariant that the target player restores every turn, rather than computing Grundy values case by case

▷ **Cue:** the parity needed for a pairing invariant is not present at the start of the game $\longrightarrow$ **Move:** find an explicit single first move that permanently removes one element from each side of the parity mismatch, reducing to the base (correct-parity) case with the opponent forced to move first into it

2020 B3

2020 B3. Let $x_0 = 1$, and let δ be some constant satisfying $0 < \delta < 1$. Iteratively, for $n = 0, 1, 2, \ldots$, a point x_{n+1} is chosen uniformly from the interval $[0, x_n]$. Let Z be the smallest value of n for which $x_n < \delta$. Find the expected value of Z, as a function of δ.

Fields: Probability (expectation, Poisson processes / order statistics of uniforms) **Position:** B3 – the third problem of the second half, an expectation calculation that admits a direct analytic solution but rewards recognizing a distributional shortcut **Answer:** $E[Z] = 1 - \ln(\delta)$.

Layer 1 THE WHAT — surface analysis

The natural attempt. The problem defines Z recursively through a conditional uniform distribution, so the direct instinct is to build a functional equation for the expectation and grind it. Let $g(\delta, c)$ be the expected value of Z given $x_1 = c$, so that $E[Z] = f(\delta) = \int_0^1 g(\delta, c)\, dc$ (since x_1 itself is uniform on $[0, 1]$). If $c < \delta$, then $Z = 1$ already, so $g(\delta, c) = 1$. If $c \geq \delta$, the process from $x_1 = c$ onward is the same process rescaled: rescaling $[0, c]$ linearly to $[0, 1]$ sends δ to δ/c, so the number of further steps needed has the same distribution as Z itself with threshold δ/c, giving $g(\delta, c) = 1 + f(\delta/c)$. This produces the exact functional equation

$$f(\delta) = \int_0^1 g(\delta, c)\, dc = \delta + \int_\delta^1 \big(1 + f(\delta/c)\big)\, dc = 1 + \int_\delta^1 f(\delta/c)\, dc.$$

This is entirely correct and is, in fact, the official route: substituting $h(x) = f(1/x)$ turns it into $h(x) = 1 + \frac{1}{x}\int_1^x h(c)\, dc$, and differentiating $xh(x) - x = \int_1^x h(c)\, dc$ gives $h(x) + xh'(x) - 1 = h(x)$, so $h'(x) = 1/x$, $h(x) = 1 + \ln x$, and $f(\delta) = 1 - \ln \delta$. It works, and it is not wrong to do it this way – but it costs a nested integral equation, a substitution to decouple the variables, a differentiation, and a constant fixed by a boundary value, all to compute the mean of a process whose underlying randomness is, on inspection, much simpler than an integral equation suggests.

The model chain. Before setting up any recursion, ask what x_n actually *is* as a random variable. Given x_n, the next value x_{n+1} is uniform on $[0, x_n]$; writing $U_{n+1} = x_{n+1}/x_n$, the ratio of a Uniform$[0, c]$ variable to c is Uniform$[0, 1]$ regardless of c (rescaling divides both the sample space and the density correctly), and this ratio is independent of x_n by construction. So $U_1, U_2, \ldots$ are i.i.d. Uniform$(0, 1)$, and

$$x_n = U_1 U_2 \cdots U_n$$

is literally a product of n i.i.d. uniforms, with $x_0 = 1$ matching the empty product. Take logarithms: $-\ln x_n = \sum_{i=1}^n (-\ln U_i)$, and each $-\ln U_i$ is Exponential(1), because for $U \sim$ Uniform$(0, 1)$, $P(-\ln U > t) = P(U < e^{-t}) = e^{-t}$ for $t \geq 0$, which is exactly the survival function of Exp(1). So $-\ln x_n = E_1 + \cdots + E_n$ is a sum of n i.i.d. Exp(1) variables, i.e. the arrival time of the n-th event of a rate-1 Poisson process.

The event $x_n < \delta$ is exactly $-\ln x_n > -\ln \delta =: t$. So Z is the smallest n such that the n-th arrival time of a rate-1 Poisson process exceeds t – equivalently, $Z = N(t) + 1$, where $N(t)$ is the number of Poisson arrivals in $[0, t]$ (the process has had exactly $N(t)$ arrivals by time t, and the very next one, the $(N(t) + 1)$-th, is the first to exceed t). Since $N(t) \sim$ Poisson(t) has $E[N(t)] = t$,

$$E[Z] = E[N(t)] + 1 = t + 1 = 1 - \ln \delta.$$

Same answer, reached without ever writing an integral equation, because the process was recognized as a disguised sum of i.i.d. exponentials the moment "uniform on [0, current value]" was read as "multiply by an independent Uniform(0,1)."

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths start from the same given: x_{n+1} is uniform on $[0, x_n]$. The natural attempt immediately encodes this as a conditional expectation and writes down $g(\delta, c) = E[Z \mid$

$x_1 = c]$ – a correct but bespoke recursive object that has to be manipulated back into something solvable. The model chain instead asks, before writing any expectation at all, "what kind of random variable is x_n itself?" – and answers it by noticing that "uniform on $[0, x_n]$" is a rescaling of a fixed distribution by the current state, which is precisely the signature of a multiplicative process.

> *The problem is handing me "x_{n+1} uniform on $[0, x_n]$" – that is not "pick a new point," it is "multiply x_n by an independent Uniform(0,1) factor," because uniformity on $[0, c]$ rescales to uniformity on $[0, 1]$ for any c; so x_n is a product of n i.i.d. uniforms, and a threshold condition on a product is a threshold condition on a sum of logs, which is something I already have a name and a mean for.*

> *Once I have $-\ln x_n$ as a sum of i.i.d. Exponential(1) variables, the question "how many terms until the sum exceeds t" is not a new calculation – it is asking for the number of arrivals of a rate-1 Poisson process up to time t, plus one, and I already know $E[N(t)] = t$ by definition of the Poisson process's rate.*

Layer 3 THE WHY

The reason "uniform on $[0, c]$" is a multiplicative statement and not merely a shape statement is definitional: the density of Uniform$[0, c]$ is $\frac{1}{c}$ on $[0, c]$, so for $X \sim$ Uniform$[0, c]$, the variable X/c has density $\frac{1}{c} \cdot c = 1$ on $[0, 1]$ – exactly Uniform$[0, 1]$, independent of c. This is not a property that needs proving case by case; it is what "uniform" *means* once you rescale. Once this is seen, the recursion $x_{n+1} = x_n \cdot U_{n+1}$ with U_{n+1} i.i.d. Uniform$(0, 1)$, independent of everything before it, is forced by the problem's own conditional definition – it was never a choice to model it this way, only a choice to notice it.

The reason $-\ln U$ is Exponential(1) is, likewise, closer to a definition than a computation: the exponential distribution with rate 1 is characterized by its survival function $P(E > t) = e^{-t}$, and $P(-\ln U > t) = P(U < e^{-t}) = e^{-t}$ matches this identically for $U \sim$ Uniform$(0, 1)$. And the reason a sum of n i.i.d. Exp(1) variables is exactly "the time of the n-th arrival of a rate-1 Poisson process" is the standard axiomatic description of a Poisson process: its defining property is that interarrival times are i.i.d. Exp(rate). So $Z - 1$, the number of full exponential steps taken before exceeding t, is not merely analogous to a Poisson count – it *is* one, by the same axiom, and $E[N(t)] = t$ is the rate-1 Poisson process's defining mean, not a fact requiring a separate derivation. Every step of the model chain is thus reading off a definition rather than performing new work; the entire content of the shortcut is refusing to re-derive facts (the mean of a Poisson count, the memorylessness implicit in "still uniform after rescaling") that are already named and known, once the disguise is removed.

This is also why the natural attempt, though correct, is more expensive: it re-derives the mean of Poisson(t) from scratch, disguised as an ODE for $h(x) = 1 + \ln x$, without ever noticing that it is doing so. The differential equation $h'(x) = 1/x$ is exactly the statement "the expected number of arrivals grows linearly in time," proved analytically instead of cited.

Caveat. None required for this problem: the answer $E[Z] = 1 - \ln \delta$ was verified by two independent methods – the official integral-equation route and the Poisson-process route derived here – and they agree exactly.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is the phrase "x_{n+1} chosen uniformly from $[0, x_n]$" – on seeing "uniform on an interval proportional to the current value," write down immediately that this is multiplication by an independent Uniform(0,1) factor, not merely "pick a new point in a shrinking interval."

2. That cue suggests taking logarithms to convert the multiplicative recursion into an additive one, landing on a sum of i.i.d. random variables – the machinery of renewal theory and Poisson processes.

3. Hesitation occurs at the step "what is the distribution of $-\ln(\text{Uniform}(0,1))$?" – it is easy to forget this is exactly Exponential(1). The reinforcement is to remember the general inverse-CDF fact: if $U \sim \text{Uniform}(0,1)$, then $F^{-1}(U)$ has distribution F for any CDF F; since $\text{Exp}(1)$ has CDF $1 - e^{-x}$ with inverse $-\ln(1-u)$, and $1 - U \sim U$ in distribution, $-\ln U \sim \text{Exp}(1)$.

4. The trap is building the full bivariate recursive functional equation $g(\delta, c)$ and grinding an ODE – it works, but it silently re-derives $E[\text{Poisson}(t)] = t$ from scratch under a different name, at the cost of a substitution, a differentiation, and fixing a constant from a boundary condition.

5. Recognisable faster next time by naming the signature: any process defined by "next value is uniform on $[0, \text{current value}]$," or more generally "next value is a fixed distribution rescaled by the current state," is a multiplicative i.i.d. process in disguise – take logs and check.

6. Reusable rule: before writing an integral or recursive functional equation for an expectation, ask whether the generating process is secretly an i.i.d. sum (via logs, differences, or ratios); if so, cite the known mean of that sum (Poisson count, negative binomial, etc.) directly instead of re-deriving it analytically.

▷ **Cue:** a recursive process defined by "next value uniform on $[0, \text{current value}]$" (or any rescaling of a fixed distribution by the current state) $\longrightarrow$ **Move:** recognize it as multiplication by an independent copy of the base distribution; take logarithms to convert the multiplicative recursion into a sum of i.i.d. random variables

▷ **Cue:** a sum of i.i.d. Exponential(rate) variables compared against a fixed threshold t $\longrightarrow$ **Move:** identify the count of terms needed to exceed t as (Poisson arrivals by time t) $+1$, and use $E[\text{Poisson}(t)] = t$ directly instead of a separate calculation

2020 B4

2020 B4. Let n be a positive integer, and let V_n be the set of integer $(2n+1)$-tuples $\mathbf{v} = (s_0, s_1, \cdots, s_{2n-1}, s_{2n})$ for which $s_0 = s_{2n} = 0$ and $|s_j - s_{j-1}| = 1$ for $j = 1, 2, \cdots, 2n$. Define

$$q(\mathbf{v}) = 1 + \sum_{j=1}^{2n-1} 3^{s_j},$$

and let $M(n)$ be the average of $\frac{1}{q(\mathbf{v})}$ over all $\mathbf{v} \in V_n$. Evaluate $M(2020)$.

Fields: combinatorics (group actions on necklaces); averaging arguments **Position:** B4 — the first genuinely hard problem on the B paper, and the one where the correct method is invisible to computation and obvious to structure **Answer:** $M(2020) = \dfrac{1}{4040}$; in fact $M(n) = \dfrac{1}{2n}$, and the base 3 may be replaced by any $a > 0$ without changing the answer.

Layer 1 THE WHAT — surface analysis

The natural attempt. Start by getting your hands dirty, which is right. V_n is the set of ± 1 walks of length $2n$ from 0 back to 0, so $|V_n| = \binom{2n}{n}$. Compute:

For $n = 1$ the two walks give $s_1 = \pm 1$, so $q = 1 + 3 = 4$ and $q = 1 + \frac{1}{3} = \frac{4}{3}$, whence

$$M(1) = \tfrac{1}{2}\left(\tfrac{1}{4} + \tfrac{3}{4}\right) = \tfrac{1}{2}.$$

For $n = 2$ the six walks give q-values $16, 8, \frac{16}{3}, \frac{16}{3}, \frac{8}{3}, \frac{16}{9}$, and

$$M(2) = \tfrac{1}{6}\left(\tfrac{1}{16} + \tfrac{2}{16} + \tfrac{3}{16} + \tfrac{3}{16} + \tfrac{6}{16} + \tfrac{9}{16}\right) = \tfrac{1}{6} \cdot \tfrac{24}{16} = \tfrac{1}{4}.$$

So $M(n) = \frac{1}{2n}$, presumably, and $M(2020) = \frac{1}{4040}$. Now prove it, and this is where the natural route stops being natural.

Induction on n requires a relation between V_n and V_{n-1}, and there isn't one that respects q: deleting a step from a walk changes *every* remaining s_j that comes after it, hence every term of q. Generating functions are worse. The quantity being averaged is $1/q$, the reciprocal of a sum of $2n$ terms; a generating function needs the summand to factor over the steps, and $\frac{1}{x_1+\cdots+x_{2n}}$ factors over nothing. You can integrate the reciprocal away, $\frac{1}{q} = \int_0^\infty e^{-qt}\,dt$, and then $e^{-qt} = \prod_j e^{-t3^{s_j}}$ does factor — but only over j, and the s_j are not independent, so the product is a transfer-matrix computation with a continuous parameter t and you are worse off than when you started.

There is one further attempt that a good solver will make, and it is worth making because it very nearly works. The map $\mathbf{v} \mapsto -\mathbf{v}$ is an involution of V_n, and it sends $q = \sum 3^{s_j}$ to $\sum 3^{-s_j}$. For $n = 1$ the pair $\{\frac{1}{4}, \frac{3}{4}\}$ sums to exactly $1 = 2 \cdot M(1)$ — the involution accounts for the whole answer. Try it at $n = 2$: the pair UUDD, DDUU contributes $\frac{1}{16} + \frac{9}{16} = \frac{10}{16}$, and $2M(2) = \frac{8}{16}$. The pairing is not the mechanism; it was an accident of $2n = 2$. That is the wall, and it is an informative one: the correct group is not of order 2.

The model chain. Two observations, in this order.

(1) The 1 is not a constant. Since $s_{2n} = 0$ we have $3^{s_{2n}} = 1$, so

$$q(\mathbf{v}) = 1 + \sum_{j=1}^{2n-1} 3^{s_j} = \sum_{j=1}^{2n} 3^{s_j}.$$

The summation range is now a full period of $2n$ consecutive indices, with no privileged endpoint.

(2) The walk is a necklace. Encode $\mathbf{v}$ by its step word $\mathbf{w} = (w_1, \ldots, w_{2n})$, $w_i = s_i - s_{i-1} \in \{\pm 1\}$; the conditions $s_0 = s_{2n} = 0$ say exactly that $\mathbf{w}$ has n entries $+1$ and n entries -1, and $\mathbf{v} \mapsto \mathbf{w}$ is a bijection from V_n onto that set W_n of words. Extend the indices cyclically, $w_{i+2n} := w_i$; since $\sum_{i=1}^{2n} w_i = 0$, the partial sums $s_j = w_1 + \cdots + w_j$ then also satisfy $s_{j+2n} = s_j$ for all $j \geq 0$, so s is a well-defined function on $\mathbb{Z}/2n\mathbb{Z}$.

Let ϕ be the cyclic shift $\phi(w_1, \ldots, w_{2n}) = (w_2, \ldots, w_{2n}, w_1)$, an action of $\mathbb{Z}/2n\mathbb{Z}$ on W_n. The partial sums of $\phi^k\mathbf{w}$ are $s_{j+k} - s_k$, so with $Q(\mathbf{w}) := \sum_{j=1}^{2n} 3^{s_j}$,

$$q(\phi^k\mathbf{w}) = \sum_{j=1}^{2n} 3^{\,s_{j+k}-s_k} = 3^{-s_k} \sum_{j=1}^{2n} 3^{s_j} = 3^{-s_k}\, Q(\mathbf{w}),$$

the middle step being a reindexing of a sum over a full cycle. In particular Q is constant on each shift orbit. Therefore

$$\sum_{k=1}^{2n} \frac{1}{q(\phi^k\mathbf{w})} = \sum_{k=1}^{2n} \frac{3^{s_k}}{Q(\mathbf{w})} = \frac{Q(\mathbf{w})}{Q(\mathbf{w})} = 1. \tag{†}$$

The average of $1/q$ over the multiset $\phi^1\mathbf{w}, \ldots, \phi^{2n}\mathbf{w}$ is thus $\frac{1}{2n}$. That list runs through the orbit of $\mathbf{w}$ exactly $2n/m$ times, where m is the least period of $\mathbf{w}$, hitting every orbit element equally often; so the average of $1/q$ over the *orbit* is also $\frac{1}{2n}$. Since W_n is the disjoint union of its orbits and every orbit has that same average, the average over all of W_n is $\frac{1}{2n}$:

$$M(n) = \frac{1}{2n}, \qquad M(2020) = \frac{1}{4040}. \qquad \square$$

Nothing used 3 beyond $3 > 0$ (so that no denominator vanishes): the same proof gives $\frac{1}{2n}$ for $q = 1 + \sum a^{s_j}$ with any $a > 0$.

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths agree through the small cases and the conjecture $M(n) = \frac{1}{2n}$. They split at the first line written after the conjecture. Call the two candidate first lines:

(N) "Let me relate V_n to V_{n-1}" — or "let me write a generating function" — both of which treat $\mathbf{v}$ as a *path*, an object with a beginning and an end, parametrised by its length n.

(M) "$3^{s_{2n}} = 1$, so $q = \sum_{j=1}^{2n} 3^{s_j}$" — which treats $\mathbf{v}$ as a *loop*, an object with no beginning, parametrised by a cyclic index set.

Every subsequent difficulty on branch (N) is caused by the basepoint: deleting a step, or conditioning on the first step, is an operation that only makes sense because you chose where the walk starts, and q does not respect that choice. Branch (M) deletes the choice instead.

> *The constraint is $s_0 = s_{2n} = 0$, so the walk returns to where it started. That is not a boundary condition on an interval, it is a statement that the index set is a circle. Before I do anything else I should write every quantity in the problem in a form where the index set is $\mathbb{Z}/2n\mathbb{Z}$ and see what survives — and the stray 1 in q is exactly $3^{s_{2n}}$, so q is already a sum over the whole circle in disguise.*

> *The answer is $\frac{1}{2n}$: one divided by the number of positions on the circle. An average that equals 1 over the size of a group is almost never a coincidence — it means the group acts on the objects and each orbit contributes a total of exactly 1. So I should stop trying to evaluate the sum and instead try to prove that $\sum_k 1/q(\phi^k \mathbf{w}) = 1$ on the nose.*

> *Reciprocals never cooperate with sums — unless the denominators in the family are all the same number up to a scalar. So the question is: does shifting the basepoint change q by a* factor*? It does: shifting by k subtracts s_k from every exponent, which divides q by 3^{s_k}. Then $1/q(\phi^k \mathbf{w}) = 3^{s_k}/Q$, and those are the terms of Q itself, normalised. The reciprocals are a probability distribution on the orbit; of course they sum to 1.*

- ▷ **Cue:** a walk or tuple with $s_0 = s_N = 0$ imposed $\longrightarrow$ **Move:** treat the index set as $\mathbb{Z}/N\mathbb{Z}$, not as $\{0, \dots, N\}$; look for the cyclic shift group before looking for a recursion in N
- ▷ **Cue:** a stray additive constant in an expression full of a^{s_j} terms $\longrightarrow$ **Move:** check whether the constant is one of the terms, i.e. a^{s_0} or a^{s_N}, and absorb it to make the summation range a full period
- ▷ **Cue:** the answer is $1/(\text{size of a natural symmetry group})$ $\longrightarrow$ **Move:** find the group action and prove each orbit contributes total 1; do not try to evaluate the sum term by term
- ▷ **Cue:** an average of reciprocals $1/q_i$ over a family $\longrightarrow$ **Move:** look for a relation $q_i = \lambda_i Q$ with Q constant on the family; then $\sum_i 1/q_i = (\sum_i \lambda_i^{-1})/Q$, and you want λ_i^{-1} to be the terms of Q
- ▷ **Cue:** a specific constant in the problem (here 3) that the answer does not mention $\longrightarrow$ **Move:** the mechanism cannot depend on that constant — prove the general-a statement, which is strictly easier because it forbids you from using arithmetic accidents

Layer 3 THE WHY

Why must the cyclic reading be forced, rather than found?

Because of what the hypothesis $s_0 = s_{2n} = 0$ *is*. A tuple with $|s_j - s_{j-1}| = 1$ and free endpoints is a path: it has a first index and a last index, and those two indices are different kinds of object from the ones in the middle. The moment you also impose $s_{2n} = s_0$, index $2n$ and index 0 carry the same data, and gluing them is not a clever re-interpretation, it is the removal of a redundancy. A function on $\{0, 1, \dots, 2n\}$ taking equal values at the two ends *is* a function

on $\mathbb{Z}/2n\mathbb{Z}$; there is nothing to discover. The definition of V_n is a definition of a necklace with n beads of each colour, written in coordinates that pretend otherwise.

Once the object is a necklace, the symmetry group is not a choice either. The bead colours w_i are intrinsic; the labelling $i = 1, \ldots, 2n$ is not. Any quantity you compute from the coordinates is a quantity computed from a choice of basepoint, so the first question about any such quantity is: how does it transform when the basepoint moves? For q the answer is that it scales, and it scales because 3^{s_j} is an *exponential* of a partial sum. Shifting the basepoint from 0 to k replaces each partial sum s_j by $s_{j+k} - s_k$; exponentials turn that subtraction into division by the single factor 3^{s_k}, uniformly in j. So q is not invariant, but it is invariant up to a scalar that depends only on the shift — and a quantity invariant up to a scalar has an invariant reciprocal-sum. The base 3 could be any positive a precisely because all that was used is $a^{x-y} = a^x / a^y$.

That last point is the real reason (†) is an identity rather than a computation: $\sum_k 1/q(\phi^k \mathbf{w}) = \sum_k 3^{s_k}/Q$ and the numerators 3^{s_k}, as k runs over the circle, are exactly the terms whose sum is Q. The orbit sum is Q/Q. You are not evaluating anything; you are recognising that the $2n$ reciprocals are the normalised weights of the very sum they came from.

Kedlaya's solution opens by asserting the more general fact for arbitrary nonzero a, then defines W_n, then ϕ, then discusses orbits before any computation appears. Here is where that comes from. Generalising 3 to a is not a flourish: it is a deliberate removal of the only handle a computational approach could have grabbed, which tells you immediately that the proof must be structural. Defining W_n before V_n is the same move as absorbing the 1 — the step word is the basepoint-free description, and $\mathbf{v}$ is the basepoint-dependent one. The orbit discussion that follows is not a technicality about periodic words; it is the statement that the averaging in (†), which is over the *list* $\phi^1\mathbf{w}, \ldots, \phi^{2n}\mathbf{w}$, descends to the *set* that is the orbit. That step is where a careless write-up loses a mark, and it costs one sentence: the list traverses the orbit $2n/m$ times with each element appearing equally often, so the two averages coincide.

One last piece of the "of course". Why should the answer not depend on n except through $2n$? Because the only quantity in the final identity that remembers anything about the walk is Q, and Q cancels. What is left is a count of basepoints. Any problem whose answer is $\frac{1}{|G|}$ for a group G acting on the configurations is telling you, in advance, that the configurations were never the point.

Caveat. I verified $M(n) = \frac{1}{2n}$ by exact rational enumeration over all $\binom{2n}{n}$ walks for $n = 1, \ldots, 5$, and verified the base-independence by recomputing $n = 3$ with $a = 5$ (both give $\frac{1}{6}$). The orbit argument I have checked by hand on the period-2 word $(+, -, +, -)$, where the orbit has size 2 and the list of $2n = 4$ shifts traverses it twice: q-values 8 and $\frac{8}{3}$, giving $2(\frac{1}{8} + \frac{3}{8}) = 1$ as (†) requires. For $a < 0$ the statement as Kedlaya gives it ("any nonzero a") requires $q(\mathbf{v}) \neq 0$, which can fail — e.g. $a = -1$ makes q an integer that vanishes for suitable walks — so I have stated the generalisation for $a > 0$ only.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is the pair of conditions $s_0 = s_{2n} = 0$ standing next to a sum that runs only from $j = 1$ to $2n - 1$ with a lone $+1$ in front. On seeing it, write two things: "the index set is $\mathbb{Z}/2n\mathbb{Z}$" and "$1 = 3^{s_{2n}}$, so $q = \sum_{j=1}^{2n} 3^{s_j}$". The asymmetric summation range is a deliberate disguise and removing it is free.

2. The cue suggests a group action: the cyclic shift by one step, acting on the $\binom{2n}{n}$ step words. The method is then orbit-averaging — prove the average of $1/q$ is $\frac{1}{2n}$ on every single orbit, and the global average follows with no counting of orbits, no Burnside, and no case analysis on periodicity beyond one sentence.

3. Hesitation occurs immediately after the small cases confirm $\frac{1}{2n}$, because the conjecture is cheap and the proof looks like it should be an induction. It is reinforced away by testing the two candidate structures against the definition of q: an induction must relate q on V_n to

q on V_{n-1}, and one line of checking shows that removing a step shifts every later exponent, so no such relation exists. Second hesitation: after finding $q(\phi^k \mathbf{w}) = 3^{-s_k} Q$, it is tempting to think the orbit might be shorter than $2n$ and to start case-splitting on the period. Don't. Average over $k = 1, \ldots, 2n$ regardless; repetition is harmless because every element repeats equally.

4. The trap is the involution $\mathbf{v} \mapsto -\mathbf{v}$. It is the first symmetry anyone sees, it is genuinely a symmetry of V_n, and at $n = 1$ it gives the answer exactly — which is the worst possible outcome, because it certifies a false method on the only case you can compute in your head. The general trap it exemplifies: a group of order 2 found immediately will crowd out a group of order $2n$ found by thinking, and the answer $\frac{1}{2n}$ was telling you the order of the group you needed all along. A secondary trap is zombie maths in its purest form — expanding $\binom{2n}{n}$ reciprocals with a transfer matrix, which is a well-defined computation that never terminates.

5. Recognisable faster next time by one reflex: *read the answer's shape before choosing a method.* If the answer to a counting-average problem is $1/N$ where N is a structural count in the problem (the length, the number of positions, the number of coordinates), assume a group of order N acts and that each orbit contributes total 1. Then spend your effort only on identifying the action. This reading also works backwards: it is how you decide, at the start, that the answer $\frac{1}{4040}$ is about the number $4040 = 2n$ of beads rather than about 2020 or about 3.

6. Reusable decision rule: *when a quantity is not invariant under the problem's symmetry group, find out how it transforms before deciding it is useless.* A quantity that transforms by a scalar — $q(gx) = \lambda(g)\, q(x)$ — is as good as an invariant, because any expression homogeneous of degree 0 in it is invariant, and because $\sum_g 1/q(gx) = \left(\sum_g \lambda(g)^{-1}\right)/q(x)$ has a chance of collapsing. Exponentials of additive data always transform this way under translation of the index, which is why sums of a^{s_j} over closed walks, character sums over shifted arguments, and products over rotated tuples all yield to the same move. The general habit: never demand invariance when equivariance will do.

2020 B5

2020 B5. For $j \in \{1, 2, 3, 4\}$, let z_j be a complex number with $|z_j| = 1$ and $z_j \neq 1$. Prove that

$$3 - z_1 - z_2 - z_3 - z_4 + z_1 z_2 z_3 z_4 \neq 0.$$

Fields: complex numbers on the unit circle; algebraic identities; trigonometric inequalities **Position:** B5 — a proof problem with no answer to find, where the entire difficulty is choosing the coordinates **Answer:** The expression never vanishes; writing P for it, the identity $P = (1 - z_1)(1 - z_2) + (1 - z_3)(1 - z_4) + (1 - z_1 z_2)(1 - z_3 z_4)$ plus half-angle coordinates reduces $P = 0$ to a pair of real equations that are incompatible with $z_j \neq 1$.

Layer 1 THE WHAT — surface analysis

The natural attempt. There are three defaults here and it is worth walking into all three, because which of them you abandon first decides how long the problem takes.

(a) Bound the modulus. $|z_1 z_2 z_3 z_4| = 1$, so it would suffice to show $|3 - z_1 - z_2 - z_3 - z_4| > 1$. Test it: put every z_j close to 1. Then $|3 - \sum z_j| \to |3 - 4| = 1$, so the inequality is false, and by continuity $|P|$ itself gets arbitrarily close to 0. This is not a near miss to be repaired with a sharper estimate. The infimum of $|P|$ over the stated domain *is* 0. Any argument that produces a positive lower bound is proving something untrue, so every triangle-inequality route is dead before it starts — which is the single most useful thing to learn in the first two minutes.

(b) Use $\bar{z} = 1/z$ and symmetric functions. Write $e_1, \dots, e_4$ for the elementary symmetric functions of $z_1, \dots, z_4$, so $P = 3 - e_1 + e_4$. If $P = 0$ then $\overline{P} = 0$; multiplying $\overline{P}$ by e_4 and using $\overline{z_j} = 1/z_j$ turns $\sum_j \overline{z_j}$ into $\sum_j e_4/z_j = e_3$, giving

$$e_1 = 3 + e_4, \qquad e_3 = 1 + 3e_4.$$

Now the goal is to prove that $f(z) = \prod_j (z - z_j)$ has 1 as a root, i.e. that $f(1) = 1 - e_1 + e_2 - e_3 + e_4 = 0$; substituting the two relations, $f(1) = e_2 - 3 - 3e_4$. So the whole problem has been reduced to proving $e_2 = 3 + 3e_4$ — and there is no further free equation. The condition "all roots on the unit circle" gives $e_4\overline{e_2} = e_2$, which is one real equation, not two, and it is not enough. The route is coherent, it uses the hypothesis honestly, and it stops one equation short. That is the wall.

(c) Optimise. Set $z_j = e^{i\alpha_j}$ and minimise $|P|^2$ over the torus. This works — it is the route Kedlaya's first solution takes — but it is four-variable calculus with a gradient condition, a factorisation of the resulting trigonometric sum, and a case analysis at the end. In the arena it is a forty-minute commitment with no guarantee the critical-point equations factor.

The model chain. The hypothesis is $z_j \neq 1$. So the quantities that are guaranteed nonzero are the four numbers $1 - z_j$, and the proof should be written in them.

Step 1: an identity. Test the claim at its own boundary. If $z_1 = z_2 = 1$ then

$$P = 3 - 1 - 1 - z_3 - z_4 + z_3 z_4 = (1 - z_3)(1 - z_4),$$

so P really is built out of such products. Try to build all of it. Since

$$(1 - z_1)(1 - z_2) + (1 - z_3)(1 - z_4) = 2 - \textstyle\sum_j z_j + z_1 z_2 + z_3 z_4,$$

the remainder is $1 - z_1 z_2 - z_3 z_4 + z_1 z_2 z_3 z_4$, which factors. Hence, exactly,

$$P = (1 - z_1)(1 - z_2) + (1 - z_3)(1 - z_4) + (1 - z_1 z_2)(1 - z_3 z_4). \tag{$*$}$$

Step 2: half-angle coordinates. Write $z_j = e^{i\alpha_j}$ with $\alpha_j \in (0, 2\pi)$ — legitimate precisely because $z_j \neq 1$. Then $1 - e^{i\theta} = -2i \sin(\theta/2)\, e^{i\theta/2}$, so for any two angles

$$(1 - e^{i\theta})(1 - e^{i\psi}) = -4 \sin\tfrac{\theta}{2} \sin\tfrac{\psi}{2}\, e^{i(\theta+\psi)/2}.$$

Put

$$A = \tfrac{\alpha_1+\alpha_2}{2}, \quad B = \tfrac{\alpha_3+\alpha_4}{2}, \quad p = \sin\tfrac{\alpha_1}{2} \sin\tfrac{\alpha_2}{2}, \quad q = \sin\tfrac{\alpha_3}{2} \sin\tfrac{\alpha_4}{2}.$$

Each $\alpha_j/2$ lies in $(0, \pi)$, so $p > 0$ *and* $q > 0$ *strictly*; this is the only place the hypothesis $z_j \neq 1$ is used, and it is used decisively. Since $z_1 z_2 = e^{2iA}$ and $z_3 z_4 = e^{2iB}$, $(*)$ becomes

$$P = -4 \left(p\, e^{iA} + q\, e^{iB} + \sin A \sin B\, e^{i(A+B)} \right).$$

Multiplying by $e^{-i(A+B)}$ and splitting into imaginary and real parts, $P = 0$ is equivalent to the pair

$$p \sin B + q \sin A = 0, \tag{i}$$

$$p \cos B + q \cos A + \sin A \sin B = 0. \tag{ii}$$

Step 3: the one constraint linking p to A. By the product formula,

$$p = \tfrac{1}{2} \left(\cos\tfrac{\alpha_1 - \alpha_2}{2} - \cos A \right) \leq \tfrac{1}{2}(1 - \cos A) = \sin^2 \tfrac{A}{2},$$

and likewise $q \le \sin^2 \frac{B}{2}$. Together with $p, q > 0$ these are the *only* facts needed; $\alpha_1 - \alpha_2$ never appears again.

Step 4: the contradiction. Suppose $P = 0$.

If $\sin A = \sin B = 0$, then $A = B = \pi$ (the only zero of sin in $(0, 2\pi)$), and (ii) reads $-p - q = 0$, impossible as $p, q > 0$.

Otherwise (i) forces $\sin A$ and $\sin B$ to be nonzero of opposite signs. Swapping the pair $\{z_1, z_2\}$ with $\{z_3, z_4\}$ leaves P unchanged, so assume $\sin A > 0 > \sin B$, i.e. $A \in (0, \pi)$ and $B \in (\pi, 2\pi)$. Put $B' = 2\pi - B \in (0, \pi)$, so $\cos B = \cos B'$, $\sin B = -\sin B'$ and $\sin \frac{B}{2} = \sin \frac{B'}{2}$. Now (i) says $p \sin B' = q \sin A$, so the number

$$t := \frac{2p}{\sin A} = \frac{2q}{\sin B'} \; > \; 0$$

is well defined, and Step 3 becomes the pair of bounds

$$t \le \tan \tfrac{A}{2}, \qquad t \le \tan \tfrac{B'}{2}. \tag{iii}$$

Substituting $p = \frac{t}{2} \sin A$, $q = \frac{t}{2} \sin B'$ into (ii) gives

$$\tfrac{t}{2} \left(\sin A \cos B' + \cos A \sin B' \right) = \sin A \sin B', \quad \text{i.e.} \quad \tfrac{t}{2} \sin(A + B') = \sin A \sin B' > 0,$$

so $\sin(A + B') > 0$, forcing $A + B' < \pi$, and

$$t = \frac{2 \sin A \sin B'}{\sin(A + B')}. \tag{iv}$$

It remains to see that (iv) contradicts (iii). Since $\tan \frac{A}{2} > 0$ and $\sin A = 2 \sin \frac{A}{2} \cos \frac{A}{2}$,

$$t > \tan \tfrac{A}{2} \iff 4 \cos^2 \tfrac{A}{2} \sin B' > \sin(A + B') \iff 2 \sin B' + \cos A \sin B' - \sin A \cos B' > 0,$$

that is, $t > \tan \frac{A}{2} \iff 2 \sin B' > \sin(A - B')$; symmetrically $t > \tan \frac{B'}{2} \iff 2 \sin A > \sin(B' - A)$. If $A \le B'$ then $\sin(A - B') \le 0 < 2 \sin B'$ and the first holds; if $A > B'$ then $\sin(B' - A) < 0 < 2 \sin A$ and the second holds. Either way t exceeds one of its upper bounds in (iii) — a contradiction. Hence $P \neq 0$. □

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths agree on the first line: $|z_j| = 1$, *so I should exploit that*. They split on what "exploit" means.

The natural attempt reads $|z| = 1$ as the *algebraic* relation $\bar{z} = 1/z$, and is then committed to symmetric functions, to the reciprocal polynomial, and to the shortfall of one equation described above. The model chain reads $|z| = 1$ as a *parametrisation*, $z = e^{i\alpha}$ with $\alpha \in (0, 2\pi)$ — and the interval is open at both ends exactly because $z \neq 1$.

That is the bifurcation, but it is the second half that does the work: having parametrised, you must group the angles as $\frac{\alpha_1 + \alpha_2}{2}$ and $\frac{\alpha_1 - \alpha_2}{2}$ rather than keeping α_1, α_2 apart, and you only know to do that if you have already rewritten P as a sum of products $(1 - z)(1 - w)$ by $(*)$. The identity comes first; the coordinates are then chosen to fit it.

> *The hypothesis is not "z_j is on the circle and somewhere else" — it is "$1 - z_j \neq 0$". Four quantities are promised nonzero and I have written an expression that does not mention any of them. Before anything else I should try to write P in terms of $1 - z_1, \ldots, 1 - z_4$.*

> *How do I find that rewriting? Look at where the conclusion is sharp. If two of the z_j equal 1, the expression collapses to $(1-z_3)(1-z_4)$ — a single product of two of my promised-nonzero quantities. So P is assembled from such products, and I should just subtract the obvious ones and see what remains. Subtract $(1-z_1)(1-z_2)+(1-z_3)(1-z_4)$; the remainder is $1-z_1z_2-z_3z_4+z_1z_2z_3z_4$, which is again a product of the same shape.*

> *The three terms of the identity involve z_1z_2 and z_3z_4, i.e. the angles $\alpha_1+\alpha_2$ and $\alpha_3+\alpha_4$. So the natural variables are the two* sums *A and B; the two* differences *are going to appear only inside the moduli. That is a reduction from four real unknowns to two plus two bounded amplitudes, and it is why this beats optimising over the whole torus.*

> *Every one of my three terms is a product $(1-u)(1-v)$, and for $|u|=|v|=1$ such a product has modulus $4|\sin\frac{\theta}{2}\sin\frac{\psi}{2}|$ and argument $\frac{\theta+\psi}{2}$ up to sign. So I know the modulus* and *the direction of each term exactly. Three known vectors summing to zero is two real equations — write them down and see whether the promised strict positivity of the two moduli can survive.*

- ▷ **Cue:** a hypothesis of the form "$z\neq c$" in a problem about a polynomial expression $\longrightarrow$ **Move:** rewrite the expression in terms of $(z-c)$; the hypothesis is only usable as a nonvanishing statement about that quantity
- ▷ **Cue:** a nonvanishing claim whose infimum of $|\cdot|$ is 0 $\longrightarrow$ **Move:** no estimate can work — the bad points are on the boundary of the domain, so the proof must be an exact identity plus a strict inequality, never a bound
- ▷ **Cue:** $|z|=1$ and you need more than $\bar z = 1/z$ $\longrightarrow$ **Move:** parametrise $z=e^{i\alpha}$ and use the half-angle factorisation $1-e^{i\alpha}=-2i\sin\frac{\alpha}{2}e^{i\alpha/2}$, which separates modulus from argument exactly
- ▷ **Cue:** a symmetric expression that you are allowed to break into pairs $\longrightarrow$ **Move:** try the pairing identity: a sum over a chosen pairing is often exactly expressible, and the loss of symmetry is recovered by a WLOG at the end
- ▷ **Cue:** a sum of complex terms whose individual moduli and arguments you know $\longrightarrow$ **Move:** set the sum to zero, rotate by the conjugate of one argument, and split into real and imaginary parts — two real equations, then hunt for the constraint the amplitudes cannot satisfy

Layer 3 THE WHY

Kedlaya's first solution begins: "let $z_1=e^{\alpha i}$, $z_2=e^{\beta i}$, $z_3=e^{\gamma i}$ and $f(\alpha,\beta,\gamma)=|3-z_1-z_2-z_3|^2-|1-z_1z_2z_3|^2$", then differentiates. Here is where that comes from — and here is why the route above is the one the problem is actually asking for.

Start with the shape of the claim. It is a *nonvanishing* statement on a domain whose closure contains zeros of the expression: $P=0$ when three of the z_j equal 1, and those points are limits of allowed points. So $|P|$ has infimum 0 on the domain and no quantitative estimate can exist. A statement like that can only be proved in one way: produce an exact algebraic identity whose terms are individually controlled, and derive a contradiction from a *strict* inequality that fails only in the limit. The hypothesis $z_j\neq 1$ must therefore enter as a strict positivity, and $p,q>0$ in Step 2 is that entry point. Everything else in the proof is bookkeeping.

Next, why $(*)$ and not some other identity. The four factors $1-z_j$ are what the hypothesis guarantees; a proof that never writes them cannot be using the hypothesis. But P is not the product $\prod(1-z_j)$ and it is not symmetric-looking in any obvious way, so the question is which combination of those factors it is. The problem itself answers this if you probe the boundary: set two variables to 1 and P becomes $(1-z_3)(1-z_4)$. A polynomial that restricts to a product of the remaining factors on that locus is, by inspection, a sum of such products, and the only

thing to determine is which. Subtracting the two obvious ones leaves a remainder in z_1z_2 and z_3z_4 alone, which factors because a polynomial of the form $1 - u - v + uv$ always does. There is no rabbit here; there is a probe, a subtraction, and a two-term factorisation.

Then why half-angles. This is the part that deserves to be driven to the definition. A complex number of modulus 1 is $e^{i\alpha}$; the chord from 1 to it is $1 - e^{i\alpha}$. Geometrically that chord subtends the angle α at the centre, so its length is $2|\sin\frac{\alpha}{2}|$ and its direction is perpendicular to the bisector — and the algebra $1 - e^{i\alpha} = e^{i\alpha/2}(e^{-i\alpha/2} - e^{i\alpha/2}) = -2i\sin\frac{\alpha}{2}\,e^{i\alpha/2}$ is that picture written down. So the half-angle is not a manipulation, it is the only natural coordinate on chords of the unit circle. Once every term of $(*)$ is a product of two chords, every term's modulus and direction are known exactly, in terms of the sums $\frac{\alpha_1+\alpha_2}{2}$ (direction) and the individual half-angles (modulus). The variables separate themselves.

And now the dimension count that explains why this beats Kedlaya's calculus. After $(*)$ and the half-angle substitution, the four unknown angles enter only as: two *directions* A, B, and two *amplitudes* p, q. The differences $\alpha_1 - \alpha_2$ and $\alpha_3 - \alpha_4$ survive nowhere except inside p and q, and there they survive only as the single constraint $0 < p \le \sin^2\frac{A}{2}$ (with equality iff $\alpha_1 = \alpha_2$). So the problem is genuinely two-dimensional with two bounded parameters, and the vanishing condition is two real equations in those four numbers: one equation, (i), ties p to q; the other, (ii), then pins t exactly. Two equations, two unknowns p, q, a unique solution, and the solution violates the box it must live in. The optimisation route sees none of this because it squares moduli before factoring, which mixes the sums and differences back together and produces the four-variable gradient computation it then has to unpick by trigonometric identities.

A last remark on the final inequality, because it is the one step that looks like an estimate in a proof I have just claimed cannot contain estimates. It is not an estimate about P; it is an exact comparison between the value of t forced by (iv) and the ceiling imposed by (iii). The comparison is decided by the sign of $\sin(A - B')$, i.e. by which of A, B' is larger — so the case split is not an artefact but the residue of the symmetry we broke when we chose the pairing $\{z_1, z_2\}, \{z_3, z_4\}$ in $(*)$. Break a symmetry in the identity, pay for it with a WLOG at the end. That is the standard exchange rate, and it is a good deal here: the symmetric approach costs four-variable calculus.

Caveat. I verified the identity $(*)$ and the reduction $P = -4(pe^{iA} + qe^{iB} + \sin A \sin B\, e^{i(A+B)})$ numerically on 2000 random quadruples each (maximum error $\sim 10^{-15}$, i.e. floating point only), and verified the final step — that at least one of $t > \tan\frac{A}{2}$, $t > \tan\frac{B'}{2}$ holds — on 10^6 random pairs (A, B') in the region $A, B' > 0$, $A + B' < \pi$, with zero violations; the analytic proof given above is complete and does not rest on that check. A random search over 2×10^5 quadruples found $\min|P| \approx 4 \times 10^{-3}$, attained with all four z_j within 0.3 of 1, confirming that the infimum is 0 and that no modulus bound can exist. This argument is my own and differs from both of Kedlaya's; I have not verified it against a third source.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is the pair of hypotheses "$|z_j| = 1$ *and* $z_j \neq 1$" — an excluded point on a circle. On seeing it, write down the four quantities $1 - z_j$ and the sentence "these are the objects promised nonzero". Then write $z_j = e^{i\alpha_j}$ with $\alpha_j \in (0, 2\pi)$ and note that the openness of the interval is the hypothesis.

2. The cue suggests rewriting the target in terms of $1 - z_j$, and the way to find that rewriting is to probe the degenerate locus: set two variables equal to 1 and see what the expression becomes. That produces the identity $(*)$, and $(*)$ then selects half-angle coordinates, because each of its three terms is a product of two chords of the unit circle.

3. Hesitation occurs twice. First at the very start, when the triangle inequality nearly works: $|z_1z_2z_3z_4| = 1$ and $|3 - \sum z_j|$ looks like it ought to exceed 1. Kill this in thirty seconds by putting all four z_j near 1 — the infimum is 0, so no estimate exists, and knowing that saves

you from returning to it. Second at the point where you have $e_1 = 3 + e_4$ and $e_3 = 1 + 3e_4$ and need $e_2 = 3 + 3e_4$: the symmetric-function route feels one line from the end for a long time. What reinforces it away is counting real equations — "all roots on the unit circle" gives you $e_4\overline{e_2} = e_2$, one real condition, and you need two.

4. The trap is symmetry. The expression is symmetric in $z_1, \dots, z_4$ and every instinct says to preserve that, which pushes you towards symmetric functions (route (b), which stalls) or towards a symmetric optimisation over the torus (route (c), which is forty minutes of calculus). The winning identity $(*)$ deliberately *breaks* the symmetry by choosing a pairing. A secondary trap is the belief that a proof of "$\neq 0$" must produce a positive lower bound; here it cannot, and time spent sharpening estimates is time spent proving something false.

5. Recognisable faster next time by this test: *is the conclusion an inequality with room, or an identity with no room?* Compute the infimum of the quantity you are told is nonzero. If the infimum is positive, estimate. If it is zero — because the excluded points are limits of allowed points — then the proof must be exact: an identity, a factorisation, a parity, a rank condition. This test takes one minute and it eliminates entire families of approach.

6. Reusable decision rule: *write the expression in the quantities the hypothesis is about, and find that rewriting by evaluating on the degenerate locus.* If you are told $x \neq a$, the proof will be about $x - a$; if you cannot see how the target involves $x - a$, substitute $x = a$ into the target and look at what survives — the residue tells you the shape of the identity you are missing. This is how to discover factorisations rather than remember them, and it applies equally to polynomial identities, to determinants that vanish on a locus, and to functional equations with excluded values.

2020 B6

2020 B6. Let n be a positive integer. Prove that

$$\sum_{k=1}^{n} (-1)^{\lfloor k(\sqrt{2}-1) \rfloor} \geq 0.$$

(As usual, $\lfloor x \rfloor$ denotes the greatest integer less than or equal to x.)

Fields: number theory (Beatty sequences, continued fractions); combinatorics on words **Position:** B6 — the hardest problem on the paper, and one where the sharpness of the bound rules out every analytic method before you start **Answer:** True, and the mechanism is that $\frac{1}{\sqrt{2}-1} = (\sqrt{2}-1) + 2$: the run-length encoding of the sign sequence reproduces the sign sequence itself, so a strong induction at scale $\sqrt{2}-1$ closes.

Layer 1 THE WHAT — surface analysis

The natural attempt. Write $\alpha = \sqrt{2} - 1 = 0.41421\ldots$, $a_k = \lfloor k\alpha \rfloor$, $\varepsilon_k = (-1)^{a_k}$, and $f(n) = \sum_{k=1}^{n} \varepsilon_k$. Compute:

$$\varepsilon_0, \varepsilon_1, \varepsilon_2, \ldots = +, +, +, -, -, +, +, +, -, -, +, +, +, -, -, +, +, -, -, -, +, \ldots$$

$$f(1), \ldots, f(24) = 1, 2, 1, 0, 1, 2, 3, 2, 1, 2, 3, 4, 3, 2, 3, 4, 3, 2, 1, 2, 3, 2, 1, 0.$$

Two facts land immediately, and the second is the important one. The sequence is not eventually periodic — the block $+, +$ at $k = 15, 16$ breaks the $+++\ --$ pattern — so no finite check will do. And $f(4) = f(24) = 0$: *the bound is attained*. There is no slack anywhere in this problem.

The default attack is analytic. Note a_k is even exactly when $\{k\alpha/2\} < 1/2$ (here $\{\cdot\}$ is the fractional part), so

$$f(n) = \#\{k \leq n : \{k\gamma\} < \tfrac{1}{2}\} - \#\{k \leq n : \{k\gamma\} \geq \tfrac{1}{2}\}, \qquad \gamma = \tfrac{\sqrt{2}-1}{2},$$

a discrepancy of the orbit of the irrational rotation by γ against the two half-circles. This is a genuine and correct reformulation, and it identifies the problem as belonging to a well-developed theory: because γ has bounded partial quotients, the three-distance theorem and the Ostrowski expansion give $|f(n)| = O(\log n)$, and one can even make the constant explicit.

And that is the wall, in the cleanest possible form. $O(\log n)$ is a two-sided bound on the magnitude. The problem asks for a one-sided bound with constant zero — an assertion about the *sign* of the discrepancy at every single n, which is attained with equality infinitely often. No estimate with any lossy step can produce it, because any inequality you weaken by even $\frac{1}{2}$ is false at $n = 4$. Equidistribution machinery is exactly the wrong tool: it is built to discard the arithmetic that this problem is entirely about.

Direct induction on n is no better: $f(n) = f(n-1) \pm 1$ and there is no rule governing the sign, so nothing propagates. Summing exponentially, $\sum_k (-1)^{a_k} = \sum_k e^{\pi i \lfloor k\alpha \rfloor}$, invites Weyl differencing, which again yields magnitude bounds and never a sign.

The model chain. Since ε_k takes only two values, describe it by its runs. Throughout, $\alpha = \sqrt{2} - 1$ and $\beta = 1/\alpha = \sqrt{2} + 1$, so that

$$\beta = \alpha + 2. \tag{$\star$}$$

Set $F(n) = \sum_{k=0}^{n} \varepsilon_k = f(n) + 1$; the claim is $F(n) \geq 1$ for all $n \geq 0$.

Step 1 (blocks). Since $0 < \alpha < 1$, the sequence a_k is nondecreasing and increases by 0 or 1 at each step, so for each $i \geq 0$ the set $\{k \geq 0 : a_k = i\}$ is a block of consecutive integers; let $c_i \geq 1$ be its size. On that block $\varepsilon_k = (-1)^i$, so the signs alternate from block to block. Counting,

$$c_0 + \cdots + c_{i-1} = \#\{k \geq 0 : k\alpha < i\} = \#\{k \geq 0 : k < i\beta\} = \lfloor i\beta \rfloor + 1,$$

the last equality because $i\beta \notin \mathbb{Z}$. Hence $c_0 = \lfloor \beta \rfloor + 1 = 3$, and for $i \geq 1$, using $(\star)$,

$$c_i = \lfloor (i+1)\beta \rfloor - \lfloor i\beta \rfloor = 2 + (\lfloor (i+1)\alpha \rfloor - \lfloor i\alpha \rfloor) \in \{2, 3\}.$$

Moreover $c_i = 3$ precisely when some integer j lies strictly between $i\alpha$ and $(i+1)\alpha$, i.e. when $i < j\beta < i+1$, i.e. when $i = \lfloor j\beta \rfloor$ for some $j \geq 1$; and $c_0 = 3$ corresponds to $j = 0$. Writing $\mathcal{B} = \{\lfloor j\beta \rfloor : j \geq 0\}$ (a strictly increasing list, so no index is counted twice),

$$c_i = 2 + \mathbf{1}[\, i \in \mathcal{B}\,] \qquad (i \geq 0). \tag{1}$$

Step 2 (the renormalisation). By $(\star)$, $\lfloor j\beta \rfloor = 2j + \lfloor j\alpha \rfloor$, so

$$(-1)^{\lfloor j\beta \rfloor} = (-1)^{\lfloor j\alpha \rfloor} = \varepsilon_j. \tag{2}$$

The Beatty sequence that records *where the long blocks are* carries the *same sign sequence* we started with.

Step 3 (block partial sums). Let $S_m = \sum_{i=0}^{m} (-1)^i c_i$, the value of F at the end of block m. By (1),

$$S_m = 2\sum_{i=0}^{m} (-1)^i + \sum_{\substack{j \geq 0 \\ \lfloor j\beta \rfloor \leq m}} (-1)^{\lfloor j\beta \rfloor}.$$

The first sum is 1 for m even and 0 for m odd. In the second, $\lfloor j\beta \rfloor \leq m \iff j\beta < m+1 \iff j < (m+1)\alpha$, so j runs over $0, 1, \ldots, J$ with $J = \lfloor (m+1)\alpha \rfloor$; by (2) the summand is ε_j. Therefore

$$S_m = 2 \cdot \mathbf{1}[\, m \text{ even}\,] + F(\lfloor (m+1)\alpha \rfloor)\,. \tag{3}$$

Step 4 (strong induction). We prove $F(n) \geq 1$ for all $n \geq 0$. Base: $F(0) = 1$. Let $n \geq 1$ and suppose $F(n') \geq 1$ for all $0 \leq n' < n$. Let m be the index of the block containing n, and set $S_{-1} = 0$. Then

$$F(n) = S_{m-1} + r(-1)^m, \qquad 1 \leq r \leq c_m,$$

where r is the number of indices of block m that are $\le n$. If m is even this gives $F(n) \ge S_{m-1} + 1$; if m is odd it gives $F(n) \ge S_{m-1} - c_m = S_m$.

Since every $c_i \ge 2$, block m begins at an index $\ge 2m$, so $n \ge 2m$, i.e. $m \le n/2$.

- $m = 0$: then $F(n) \ge S_{-1} + 1 = 1$.
- $m \ge 2$ even: by (3) with $m-1$ odd, $S_{m-1} = F(\lfloor m\alpha \rfloor)$, and $\lfloor m\alpha \rfloor \le \frac{n}{2}\alpha < n$, so the induction hypothesis applies and $F(n) \ge S_{m-1} + 1 \ge 2$.
- m odd: by (3), $S_m = F(\lfloor (m+1)\alpha \rfloor)$, and $\lfloor (m+1)\alpha \rfloor \le (\frac{n}{2} + 1)\alpha < n$ (since $n(1 - \frac{\alpha}{2}) > \alpha$ holds for every $n \ge 1$, as $1 - \frac{\alpha}{2} > \alpha$), so $F(n) \ge S_m \ge 1$.

In every case $F(n) \ge 1$, so $f(n) = F(n) - 1 \ge 0$. □

Layer 2 THE HOW — the bifurcation

Bifurcating point. Both paths agree on the first move — compute the sequence, see the runs of $++\!+$ and $--$ — and split on the very next line, over what kind of object $\lfloor k\alpha \rfloor$ is.

The natural attempt reads $\lfloor k\alpha \rfloor$ as *$k\alpha$ minus a bounded error*. That is the analytic reading: the sequence becomes an equidistributed orbit, the sum becomes a discrepancy, and the tools available produce two-sided magnitude bounds.

The model chain reads $\lfloor k\alpha \rfloor$ as *the block index of k*, i.e. it throws away k and keeps only the run-length word $c_0 c_1 c_2 \cdots = 3, 2, 3, 2, 3, 2, 2, 3, \ldots$. That is the combinatorial reading, and it is the one that survives, because the run-length word of this particular sequence *is* the sequence again.

Everything after the split follows: on the analytic branch you spend the paper sharpening a bound that cannot be sharpened to zero; on the combinatorial branch you spend it verifying $(\star)$ and doing the bookkeeping in Step 3.

> *Before choosing a method, test the bound for slack: $f(4) = 0$. The inequality is attained, so every step of my argument has to be exact. That single computation eliminates estimation, equidistribution, exponential sums and averaging in one stroke — whatever the proof is, it is an identity, and for $\sqrt{2}$ that means the continued fraction.*

> *The sign sequence takes two values, so all the information is in the lengths of the runs. The run lengths are 2 or 3 because their average is $1/\alpha = 2.414$ and $\alpha \in (\frac{1}{3}, \frac{1}{2})$. So compress: the alternating sum of the run lengths is exactly the quantity I am being asked about, evaluated at the block boundaries.*

> *Which runs are long? The block boundaries are at $\lfloor i\beta \rfloor$ with $\beta = 1/\alpha$, so $c_i - 2$ is the indicator of the Beatty sequence of β. Now $\beta = \sqrt{2} + 1$ and $\alpha = \sqrt{2} - 1$ differ by exactly 2 — an even integer — so $\lfloor j\beta \rfloor$ and $\lfloor j\alpha \rfloor$ have the same parity, and the alternating sum of my run lengths is literally the original sum at a smaller index. The problem contains itself.*

> *If the problem contains itself at scale $\alpha < \frac{1}{2}$, then strong induction is the right frame, not ordinary induction: I need F at index roughly $n\alpha$, which is well below n, so there is nothing delicate about the descent. The only thing I must check is that the block containing n has index at most $n/2$, which follows from every run having length at least 2.*

▷ **Cue:** a bound that is attained (test the first few cases and find equality) ⟶ **Move:** no estimate can work — look for an exact identity, a self-similarity, or an invariant; discard every $O(\cdot)$ tool immediately

- ▷ **Cue:** $\lfloor k\alpha \rfloor$ with α a quadratic irrational $\longrightarrow$ **Move:** compute $1/\alpha$ and compare it with α; a relation $1/\alpha = \alpha + N$ makes the sequence self-similar under renormalisation by α
- ▷ **Cue:** a sequence taking only two values $\longrightarrow$ **Move:** encode it by run lengths and ask what sequence the run lengths form; for Sturmian/Beatty words this is the same sequence again
- ▷ **Cue:** run lengths of $\lfloor k\alpha \rfloor$-type data $\longrightarrow$ **Move:** the block boundaries are the Beatty sequence $\lfloor i/\alpha \rfloor$; the number of k with $\lfloor k\alpha \rfloor < i$ is $\lfloor i/\alpha \rfloor + 1$
- ▷ **Cue:** a self-similar structure with contraction factor $\lambda < 1$ $\longrightarrow$ **Move:** use strong induction on n with the hypothesis applied at index $\approx \lambda n$; check only that the reduced index is genuinely smaller
- ▷ **Cue:** an alternating sum over blocks of alternating sign $\longrightarrow$ **Move:** partial sums inside a block are bracketed by the two neighbouring block-boundary sums, so bounding S_m for all m bounds every partial sum

Layer 3 THE WHY

Why is this true, and why is it true for $\sqrt{2}-1$ rather than for irrationals in general?

Start from the sharpness. $f(4) = 0$ is not an accident of small numbers: $f(24) = 0$ as well, and equality recurs. A statement whose bound is attained infinitely often is a statement about exact structure, and the only exact structure available in $\lfloor k\alpha \rfloor$ is the continued fraction of α. So the shape of the proof is forced before any of its content is known. This is the same principle as in any sharp inequality — equality cases tell you the method — but here it does something stronger: it tells you the *tool*, because the theory of $\lfloor k\alpha \rfloor$ splits cleanly into the approximate part (equidistribution, discrepancy, exponential sums) and the exact part (Beatty sequences, three-distance, Ostrowski, continued fractions), and sharpness lives entirely on the exact side.

Now the mechanism. $\alpha = \sqrt{2}-1$ has the purely periodic continued fraction $\alpha = [0; 2, 2, 2, \dots]$, equivalently $\beta := 1/\alpha$ satisfies $\beta = 2 + 1/\beta$, equivalently $(\star)$: $\beta = \alpha + 2$. Geometrically this says the rotation by α renormalises to itself: zoom in by the factor α and you see the same rotation, not a different one. That is what makes a self-similar proof possible at all, and it is available for every quadratic irrational.

But self-similarity alone is not enough, and here is the axiom-level point of the problem. The object being renormalised is not the rotation; it is the *sign* sequence $(-1)^{\lfloor k\alpha \rfloor}$, and signs care about parity. The renormalisation replaces α by β in the exponent — the long blocks sit at $i = \lfloor j\beta \rfloor$ — so the new sign sequence is $(-1)^{\lfloor j\beta \rfloor}$, and this equals the old one if and only if $\beta - \alpha$ is an *even* integer. For $\sqrt{2}-1$, $\beta - \alpha = 2$. That is the whole problem.

It is worth seeing this fail. Take $\alpha = \frac{\sqrt{5}-1}{2} = 0.618\dots$, the other famous quadratic irrational, where $1/\alpha = \alpha + 1$: the difference is odd, the renormalisation flips every sign, and the analogous partial sums are *not* nonnegative — the sum $\sum_{k \le n} (-1)^{\lfloor k\alpha \rfloor}$ reaches -8 by $n = 37512$. Nothing about the golden ratio is worse-behaved than $\sqrt{2}$; the two are equally well approximable and equally self-similar. The single bit that differs is the parity of $1/\alpha - \alpha$. Kedlaya's remark that the same proof works with $\sqrt{2}-1$ replaced by $\sqrt{n^2+1} - n$ is exactly this observation: for those numbers $1/\alpha = \alpha + 2n$, again even.

That reframes the problem's phrasing. "$\sqrt{2}-1$" looks like decoration, one specific irrational chosen to make a concrete question. It is not. It is the hypothesis, and it is doing one job: making $1/\alpha - \alpha$ an even integer so that the sign sequence is a fixed point of its own renormalisation rather than an anti-fixed point. Once you know that, the proof writes itself, because a fixed point of a contraction is exactly the setting where strong induction with no error term works.

Two smaller "of course"s inside the argument. First, why do the run lengths have to be 2 or 3? Because the average run length is $1/\alpha = 2.414$ and $\lfloor k\alpha \rfloor$ increases by 0 or 1, so runs have length $\lfloor \beta \rfloor$ or $\lceil \beta \rceil$ — the three-distance theorem in its simplest instance. Second, why is it enough to control the sums S_m at block boundaries? Because within a block the sign is constant, so the partial sums march monotonically from one boundary value to the next; a monotone excursion between two controlled endpoints needs no separate argument, and that is what

turns a statement about every n into a statement about every m — with $m \le n/2$, which is where the descent gets its room.

Kedlaya's first solution sets up the same c_i, derives $c_i = 2 + \lfloor (i+1)\alpha \rfloor - \lfloor i\alpha \rfloor$, and then proves the inequality by an induction of the form "if it holds up to N then it holds up to $4N$". Here is where that comes from: it is the same renormalisation, with the descent packaged as a doubling-up rather than as a strong induction. I have used the strong-induction form because it makes the contraction factor visible — you need F at index $\lfloor (m+1)\alpha \rfloor$, and everything else is bookkeeping about which block n sits in.

Caveat. I verified $f(n) \ge 0$ by direct computation for all $n \le 3 \times 10^5$ (minimum 0, attained first at $n = 4$), and verified the block formula (1) and the parity identity (2) over the same range. The golden-ratio failure quoted above is a computation, not a theorem: I checked that $\sum_{k \le n} (-1)^{\lfloor k(\sqrt{5}-1)/2 \rfloor}$ attains -8, which suffices to show the statement is false there. One warning against over-reading my WHY: the condition "$1/\alpha - \alpha$ is an even integer" is what makes *this proof* work; it is sufficient, not necessary for the conclusion. Numerically, $\alpha = \sqrt{3} - 1$ also appears to give nonnegative partial sums up to 3×10^5 despite $1/\alpha - \alpha \notin \mathbb{Z}$, so the exact class of α for which the inequality holds is not settled by anything written here.

Layer $+\alpha$ BACKTESTING

1. The first useful cue is $\sqrt{2} - 1$ appearing inside a floor function. On seeing it, write down $\frac{1}{\sqrt{2}-1} = \sqrt{2} + 1$ and then the line $\frac{1}{\alpha} = \alpha + 2$. Also write out twenty terms of the sign sequence and the partial sums, and *record whether the bound is attained*. Those two lines — the reciprocal identity and the value $f(4) = 0$ — between them determine the entire method.

2. The cue suggests run-length encoding plus renormalisation: compress the two-valued sequence into its block lengths $c_i \in \{2, 3\}$, identify the long blocks with the Beatty sequence of $1/\alpha$, and use $\frac{1}{\alpha} = \alpha + 2$ to recognise that the alternating sum of block lengths is the original sum at a smaller index. Then strong induction, with contraction factor $\alpha < \frac{1}{2}$.

3. Hesitation occurs at the reformulation as a discrepancy, because it feels like progress: the problem becomes a recognised object with a literature. It is progress in the wrong direction, and what reinforces it away is the equality case. Ask of any tool: *can this produce a bound with constant exactly zero, at every n?* Discrepancy theory cannot, because it is invariant under changes that destroy equality. A second hesitation comes in Step 3, at the change of summation variable from blocks i to Beatty indices j; it feels like a trick. It is bookkeeping — the sum $\sum_i (-1)^i \mathbf{1}[i \in \mathcal{B}]$ is a sum over the elements of $\mathcal{B}$ and nothing else, and the range condition $\lfloor j\beta \rfloor \le m$ unwinds to $j < (m+1)\alpha$ in one line.

4. The trap is that the problem is genuinely a discrepancy problem and the analytic theory genuinely applies. You will be able to prove $|f(n)| = O(\log n)$, which is a real theorem, is harder than what is being asked, and does not imply it. Losing an hour this way is the standard outcome for a well-read solver — the better your background, the more expensive this trap is. A second trap is trying to establish eventual periodicity from the first dozen terms: the pattern $+++\ --$ repeats three times and then breaks, so a rushed solver may "prove" periodicity and hand in a false argument.

5. Recognisable faster next time by one reflex: *compute the equality case before choosing a tool.* If a claimed inequality is attained, the admissible methods are exact ones — identities, invariants, self-similarity, parity, explicit bijections — and every asymptotic tool is disqualified regardless of how well it fits the problem's surface. Do this check in the first two minutes, in writing, because the cost of not doing it is measured in hours, not minutes.

6. Reusable decision rule: *when a problem is built from $\lfloor k\alpha \rfloor$ for a specific irrational α, the specific value of α is a hypothesis, not a decoration — find out what it is doing.* Compute $1/\alpha$ and its

continued fraction first. A relation $1/\alpha = \alpha + N$ says the structure renormalises to itself at scale α; the parity of N then says whether any sign attached to it renormalises to itself or to its negative. More generally: whenever a problem names a particular constant, locate the one identity that constant satisfies which no nearby constant satisfies, and expect that identity to be the proof. The same habit cracks problems about e (its continued fraction), about $\pi^2/6$, and about specific integer moduli chosen so that a residue class is a square.