Congruence Arguments Proving Most Integers aren't Sums of Two Squares or Two Fibonaccis

Qixuan Huang[a] and Steven J. Miller[b]

[a]Guangzhou Suyuan School, Guangzhou, China; [b]Williams College, Williamstown, MA, USA

ARTICLE HISTORY
Compiled July 10, 2026

Article type: Mathematical Outreach

ABSTRACT
Throughout the centuries, a common problem in mathematics is the following: given a set, what numbers are sums of a fixed number elements of that set? We focus on two summands, with examples ranging from writing n^{th} powers as the sum of two n^{th} powers (Fermat's Last Theorem, proved by Wiles) to representing every even number at least 4 as the sum of two primes (the Binary Goldbach problem, still open). Most integers are not the sum of two squares. The simplest proof that a positive percentage are missed is through well-known congruence arguments; we explore how far these ideas can be *elementarily* pushed, and compare the strength of their results and the ease of their application to certain analytic methods.

In particular, we consider what fraction of integers at most x are the sum of two squares or two Fibonacci numbers. By choosing an appropriate modulus and looking at the reduction of sums, we prove that more than 50% of integers are missed in both cases; in other words, we have a simple, elementary calculation showing most integers don't have such representations. Looking at moduli up to 10,000 our best results use 7056 (showing at most 38.4% of integers are sums of two squares) and 9349 (showing at most 3.34% of integers are sums of two Fibonaccis). Our result is significantly worse than what simple analytic methods give for the Fibonacci case, but exceeds the best possible bounds from lattice point counting for sums of squares.

We discuss several conjectures about how far these arguments can be pushed and which moduli are the best candidates to check. These are ideally suited to collaboration with AI. We discuss that process, highlighting how simple prompts helped us discover known results in the literature (with a slightly different emphasis than what we were doing, and using some non-elementary inputs). We hope to excite readers both in studying problems such as these, and in how to use the rapidly developing AI tools to supplement one's research.

KEYWORDS
Sum of two squares; sum of two Fibonacci numbers; congruence arguments

1. Introduction

One reason number theory questions have fascinated so many for so long is that the problems are easy to state, and in many cases elementary methods yield good answers. One of the most natural questions we can ask about a set of numbers S is what numbers can be expressed as the sum of elements from that set. Explicitly, given

Email address: sjm1@williams.edu

a set S and a positive integer k, classify all n such that

$$n = s_1 + s_2 + \cdots + s_k \quad \text{where } s_i \in S. \tag{1.1}$$

Sometimes we may place conditions on n, for example we might require n to be in the set S as well. Examples of this go back millenia to Euclid's *Elements* and the Pythagorean Theorem, which gave explicit formulas for when a square is the sum of two squares ($a^2 + b^2 = c^2$). Replacing the exponent 2 with higher powers led to Fermat's Last Theorem. Pierre de Fermat [Fe] famously claimed to have a proof in 1637 that no positive integer solutions exist for exponents $n > 2$, but it took over three centuries of false starts and partial breakthroughs before it was proved in 1995 [TW, Wi].

Similarly, Christian Goldbach's 1742 letter to Euler asked whether all integers could be written as sums of a fixed number of primes [Go]. It is here that we first see congruence arguments emerge. Consider the ternary problem, writing integers as the sum of three primes. If two of the primes are 2 (the only even prime) then the only numbers with such a representation are four more than a prime, and by the Prime Number Theorem [MT-B, SS] to first order there are only $x/\log x$ primes up to x; thus in the limit almost no numbers have such a representation. If exactly one of the primes is 2 then the other two primes are both odd and thus the sum is even; thus no odd number can be written as a sum of three primes where only one is 2. Similarly if we look at the binary problem, writing integers as a sum of two primes, if exactly one prime is 2 then we cannot get any even integer. This led to two problems: the ternary (every odd integer at least 7 is the sum of three primes) and the binary (every even integer at least 4 is the sum of two primes) Goldbach conjectures. The ternary case is understood. Ivan Vinogradov proved in 1937 that every sufficiently large odd integer is the sum of three primes [Vi]. Over the years "sufficiently large" was lowered again and again, until finally Harald Helfgott, mixing theory and computation, proved it holds for all odd numbers $n \geq 7$ [He]. However, the binary problem remains very much open.

Another example is Waring's problem (1770) on sums of k^{th} powers [Na, Wr]. Given any k, there is a number $s(k)$ such that all sufficiently large integers are the sum of at most $s(k)$ positive k^{th} powers. Adding 1^k enough times we can handle the finitely many numbers that do not have such a decomposition, proving all positive integers are the sum of at most $\widetilde{s}(k)$ perfect k^{th} powers. Sometimes it's easier to study a related problem; instead of studying sums we can study sums and differences; this is the signed Waring problem, but it introduces its own complications (if k is odd $m^k - m^k = 0$ and there can be infinitely many solutions). Additionally, if we can take the k^{th} power of negative numbers, it's no longer a finite computation for odd k as to whether or not a number has a representation; it was only recently that Andrew Booker finally answered the question: is 33 the sum of three cubes? Yes! We have

$$\begin{aligned} 33 \;=\; & (8,866,128,975,287,528)^3 \;+\; (-8,778,405,442,862,239)^3 \\ & + (-2,736,111,468,807,040)^3; \end{aligned} \tag{1.2}$$

see [Pa] for a nice account of the story, and how the presence of negative signs greatly increases the difficulty.

While many of these problems are easy to state, solving them often requires advanced mathematics or, as in the representation of 33 as the sum of three cubes, tremendous amounts of computational time and efficient algorithms. We consider

some of these classical problems and see how far we can get using strictly elementary techniques. In particular, we investigate what can be achieved using *congruence arguments*. Typically these arguments are only used to prove a positive percentage of the time something cannot happen; we see how far we can push these results, and end with several conjectures which we invite interested readers to explore with us.[1]

Congruence arithmetic, often called clock arithmetic, is well known and easy to use. We say a is equivalent to b modulo n, written as

$$a \equiv b \bmod n, \tag{1.3}$$

if $a - b$ is a multiple of n.

Despite its simplicity, it can frequently yield decent results. For example, in many introductory math classes students are shown how to use congruence arguments to easily prove that a positive percentage of numbers cannot be written as a sum of two squares [We2]. By looking at squares modulo 4, we see that any square can only have a remainder of 0 or 1 modulo 4:

$$0^2,\ 2^2 \equiv 0 \bmod 4, \quad 1^2,\ 3^2 \equiv 1 \bmod 4. \tag{1.4}$$

Thus, when we add two squares together, the remainders modulo 4 are just 0, 1 and 2 as

$$0+0 = 0, \quad 0+1 = 1, \quad 1+1 = 2. \tag{1.5}$$

As it's impossible to have a sum of two squares have a remainder of 3 modulo 4, we see at least 25% of integers (those congruent to 3 modulo 4) cannot be written as a sum of two squares.

As remarked, our goal is to see how far we can easily push these elementary arguments. Perhaps the simplest question to ask, now that we know at least a quarter of integers are not sums of two squares, is whether a majority of numbers are not sums of two squares. Equivalently, can we push the 25% above to something exceeding 50%?

Much is known about sums of squares (see for example [HW, MT-B]). While every positive integer n is the sum of four non-negative squares, for n to be the sum of two squares the prime factorization of n cannot have any primes congruent to 3 modulo 4 to an odd power. Thus neither 3 nor 7 can be written as the sum of two squares, but $3^2 7^2 13 = 5733$ can, it's $42^2 + 63^2$.

As the proof of these theorems use more advanced mathematics, we start with a simpler question where we can easily prove what happens in the limit: sums of two Fibonacci numbers $\{F_n\}$, where $F_1 = 1, F_2 = 2$ and $F_{n+1} = F_n + F_{n-1}$ for $n \geq 2$. Because the Fibonacci sequence grows exponentially fast, it's easy to show in the limit that zero percent of integers are the sum of two Fibonaccis. We use congruence arguments to prove that this percentage is less than 50%.

Theorem 1.1. *A majority of positive integers are not the sum of two Fibonacci numbers. Looking at moduli at most 10,000 we find at most about 3.34% are (coming from the modulus 9349).*

We use the techniques from the above problem to tackle the original question, and use congruence arguments to show most numbers are not the sum of two squares.

[1] Email sjm1@williams.edu if interested.

Theorem 1.2. *A majority of positive integers are not the sum of two squares. Looking at moduli at most 10,000 we find at most about 38.4% are (coming from the modulus 7056).*

The remainder of this paper is organized as follows. After reviewing some properties about Fibonacci numbers, in Section 2.2 we prove our results on sums of Fibonacci numbers. We then prove analogous results for sums of squares in Section 3. In each case we compare our results to what is obtainable through analytic arguments. While our best bounds from congruence arguments are significantly worse for sums of Fibonacci numbers, they are better than what is obtainable through lattice point counting for sums of squares, and require significantly less mathematical input (but more patience in examining candidate moduli). We conclude in Section 4 with a discussion of several conjectures and future work inspired by data, and discuss the challenges and opportunities in using AI to tackle such questions.

2. Sums of Two Fibonacci Numbers

.

2.1. *Properties of the Fibonacci Numbers*

We record some standard facts about the Fibonacci numbers $\{F_n\}$, see for example [Ko]. Recall we defined them by $F_1 = 1, F_2 = 2$, and $F_{n+2} = F_{n+1} + F_n$. The reason we start 1, 2 and not 0, 1, 1 or 1, 1 is this leads to the following result.

Theorem 2.1 (Zeckendorf's Theorem [Ze])**.** *Every positive integer has a unique representation as a sum of non-adjacent Fibonacci numbers, where no index is repeated.*

Proof. We first prove a representation exists, and then show it's unique. We proceed by induction and use the Greedy Algorithm.

The base case is trivial, as $1 = F_1$ (we also have $2 = F_2, 3 = F_3$ and $4 = F_3 + F_1$).

We now assume every integer at most m has a unique representation, and consider $m+1$. Let F_{ℓ_1} be the largest Fibonacci number at most $m+1$. If $m+1 = F_{\ell_1}$ we are done, if not by the inductive assumption $(m+1) - F_{\ell_1}$ has a unique representation as the sum of non-adjacent Fibonacci numbers, and let F_{ℓ_2} be the largest Fibonacci number in that decomposition:

$$m + 1 - F_{\ell_1} \;=\; F_{\ell_2} + \cdots + F_{\ell_r} \tag{2.1}$$

for some r. Moving F_{ℓ_1} to the other side we have a decomposition for $m+1$; it's valid so long as $\ell_1 > \ell_2 + 1$.

Clearly $\ell_1 \geq \ell_2$ as $m + 1 > m + 1 - F_{\ell_1}$. We show if ℓ_1 equals $\ell_2 + 1$ or ℓ_2 this leads to F_{ℓ_1+1} is less than $m+1$, contradicting F_{ℓ_1} being the largest Fibonacci number at most that.

- If $\ell_1 = \ell_2 + 1$ then since $m + 1 = F_{\ell_1} + F_{\ell_2} + \cdots$ we have $F_{\ell_1} + F_{\ell_1-1} = F_{\ell_1+1}$ is at most $m + 1$.

- If $\ell_1 = \ell_2$ we similarly find $F_{\ell_1} + F_{\ell_1}$ is at most $m + 1$, but this is $F_{\ell_1} + F_{\ell_1-1} + F_{\ell_1-2} = F_{\ell_1+1} + F_{\ell_1-2}$.

Thus in both cases F_{ℓ_1+1} is at most $m+1$, contradicting F_{ℓ_1} is the largest Fibonacci number at most $m+1$.

Uniqueness follows from a simple induction argument. Direct calculation handles the base case, namely the only way we can write 1 as such as sum is F_1. For the inductive step assume every integer at most m has a unique decomposition. Assume $m+1$ has two representations:

$$m+1 \;=\; F_{\ell_1} + \cdots + F_{\ell_r} \;=\; F_{j_1} + \cdots + F_{j_s} \tag{2.2}$$

with $\ell_1 > \ell_2 > \cdots$ and $j_1 > j_2 > \cdots$. If $\ell_1 = j_1$ then $(m+1) - F_{\ell_1}$ is at most m and has two decompositions, contradicting the inductive assumption. Thus without loss of generality we may assume $\ell_1 > j_1$. The smallest the ℓ-sum can be is F_{ℓ_1} while the largest the j-sum can be is $F_{j_1} + F_{j_1-2} + F_{j_1-4} + F_{j_1-6} + \cdots$. If we add 1 to the j-sum we get F_{j_1+1}; as this is at most F_{ℓ_1} we see the j-sum is smaller than the ℓ-sum, and thus these cannot be two decompositions of the same number. □

For another proof, using combinatorial arguments, see [KKMW] (this approach turns out to be very useful for examining related questions).

Theorem 2.2. *For any positive integer m, the Fibonacci sequence modulo m is periodic. The length of the repeating cycle is called the Pisano period, and is denoted by $\pi(m)$.*

Proof. This follows from the Pigeonhole Principle: for $1 \le i \le m^2+1$ look at the pairs $(F_i \bmod m, F_{i+1} \bmod m)$. As there are only m^2 pairs of integers modulo m at least two of these m^2+1 pairs are the same, and thus the sequence repeats (and once it repeats it will continue repeating, as all subsequent values are determined once two adjacent terms are chosen). □

Thus, $F_{n+\pi(m)} \equiv F_n \bmod m$ for all n. This periodicity allows us to determine exactly which remainder classes are represented by Fibonacci numbers modulo m.

Finally, we state the well-known explicit formula for the n^{th} Fibonacci number.

Theorem 2.3 (Binet's Formula)**.** *Let $F_1 = 1, F_2 = 2$ and $F_{n+1} = F_n + F_{n-1}$ for $n \ge 2$. Then*

$$F_n \;=\; \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^{n+1} - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^{n+1}. \tag{2.3}$$

There are many proofs; perhaps the simplest is the Method of Divine Inspiration: given the proposed formula, show it works for n equals 1 and 2, and show it satisfies the same recurrence as the Fibonacci sequence. Thus the two sequences must be identical!

2.2. *Upper Bound for Sums of Two Fibonacci Numbers*

We can now prove our main result, Theorem 1.1, namely that most integers are not the sum of two Fibonacci numbers.

We use Pisano periods to tackle this problem. It suffices to analyze sums of two remainders within a single repeating cycle; that tells us all the possible remainders modulo N. The following procedure generates an upper bound for the percentage of integers that are the sum of two Fibonacci numbers.

(1) Choose an integer N, and calculate the remainders of the Fibonacci sequence modulo N until the sequence starts repeating. This is a finite calculation by Theorem 2.2. We record this ordered sequence of remainders as the set $\{R_N\}$. For example, when $N = 3$, the Fibonacci remainders modulo 3 form the cycle

$$0,\ 1,\ 1,\ 2,\ 0,\ 2,\ 2,\ 1,\ 0,\ 1,\ 1,\ 2,\ 0,\ 2,\ 2,\ 1,\ 0,\ 1,\ 1,\ \ldots$$

which is periodic with length $\pi(3) = 8$.

(2) Remove any duplicate remainders from $\{R_N\}$, keeping only the unique values to form a new set $\{U_N\}$. When $N = 3$ we have $\{U_N\} = \{0, 1, 2\}$.

(3) For every pair of numbers from $\{U_n\}$ (they can be the same, and without loss of generality we may assume the first element is at most the second), add them and take the result modulo N. Collect these sums into a final set, $\{S_N\}$. For instance, if $N = 3$ there are four pairs (0,0), (0,1), (1,1), (1,2), and (2,2), and their sums modulo 3 give us the set $\{S_N\} = \{0, 1, 2\}$. Note in this case no numbers are excluded as all possible remainders modulo 3 are realized.

(4) The cardinality of S_N, denoted $|S_N|$, gives an upper bound for the percentage of remainders that *have a chance of being expressible as the sum of two Fibonacci numbers*: $\frac{|S_N|}{N} \cdot 100\%$.

We wrote a computer program to execute the above procedure and calculate these upper bounds for the percentages of numbers that can be the sum of two Fibonacci numbers. The program was run on the following system: AMD Ryzen 7 5700U (1.80 GHz) with Radeon Graphics, 16.0 GB physical RAM (13.9 GB usable) and 64-bit x64 operating system. It took 1420 seconds to compute up to 10,000. Table 2.1 displays some of our results for N at most 10,000; in the interest of space we just list the values of N where the percentage drops to a new record low (which we call "record-minimum ratio numbers"), along with their prime factorizations.

N	Percentage	Factorization of N
1	100.00%	1
29	93.10%	29
55	78.18%	5×11
76	68.42%	$2^2 \times 19$
144	51.39%	$2^4 \times 3^2$
199	43.22%	199
377	30.24%	13×29
521	24.76%	521
987	16.51%	$3 \times 7 \times 47$
1364	13.27%	$2^2 \times 11 \times 31$
2255	12.77%	$5 \times 11 \times 41$
2584	8.55%	$2^3 \times 17 \times 19$
3571	6.78%	3571
6119	6.39%	29×211
6765	4.26%	$3 \times 5 \times 11 \times 41$
9349	3.34%	9349

Table 2.1. Values of N up to 10,000 with record-minimum percentages for the sum of two Fibonacci numbers.

Interestingly, it's not until $N = 29$ that our bound is less than 100%. The first N

proving Theorem 1.1 is 199, where the percentage finally dips below 50%. The best N in this region is 9349, giving about 3.34%.

We'll return to this table in Section 4, and content ourselves here in remarking that there is no obvious pattern among the prime factors of these "record-minimum ratio numbers," or where the next such number occurs.

2.3. *Comparison with Analytic Methods*

We now compare the bounds obtained from congruence arguments with those from analytic arguments; in the case of Fibonacci numbers the analytic arguments give significantly better results, but they do have greater inputs (we use Binet's Formula and Zeckendorf's Theorem).

Consider all integers that are the sum of two Fibonacci numbers, with largest summand F_N. The number of such pairs is $\binom{N}{2}+N$ (the first term comes from choosing two distinct numbers, the second from choosing the same number twice). If we add to this the number that are the sum of one Fibonacci number, which is N, we obtain the number of integers that are the sum of at most two Fibonacci numbers with largest summand F_N is at most $\binom{N}{2} + 2n = N(N+3)/2$.

Remark 2.4. *We'll see shortly there's no need to decrease the bound above, but we can reduce it a bit using Zeckendorf's Theorem, Theorem 2.1. If we have two adjacent Fibonacci numbers we can replace their sum with the next Fibonacci number, while if we have two copies of the same Fibonacci number we can replace it with the sum of one index larger and two indices smaller. Further, two different sums cannot be the same number as Zeckendorf's theorem yields different decompositions are different numbers.*

All of these sums are at most $2F_N$; any number at most F_N that is the sum of at most two Fibonacci numbers has its largest summand at most F_N. Thus the number of integers up to F_N that are the sum of at most two Fibonacci numbers is at most $N(N+3)/2$ (it is of course less than this, as many of these numbers exceed F_N).

Thus the percentage of numbers up to F_N that can be represented as a sum of at most two Fibonacci numbers is at most $(N(N+3)/2)/F_N$. By Binet's formula (Theorem 2.3), the N^{th} Fibonacci number is

$$F_N \;=\; \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^{N+1} - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^{N+1}. \tag{2.4}$$

As $|(1-\sqrt{5})/2| < 1$ the second term above tends to zero exponentially quickly, and thus

$$F_N \;\approx\; \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^{N+1}. \tag{2.5}$$

Thus the percentage of numbers up to F_N that could be the sum of at most two Fibonacci numbers is bounded by approximately

$$\frac{N(N+3)/2}{\left(\frac{1-\sqrt{5}}{2}\right)^{N+1}/\sqrt{5}}.$$

The polynomial growth in the numerator is dwarfed by the exponential decay in the denominator, proving that in the limit 0% of integers are the sum of at most two Fibonacci numbers.

Remark 2.5. *Using Zeckendorf's theorem, we can show that in the limit 0% of integers are the sum of at most k* distinct *Fibonacci numbers; with a little work one can relax the distinctness constraint.*

3. Sum of Two Squares

We now turn to analyzing sums of two squares. Interestingly, the *elementary* congruence arguments here turn out to be far superior to the analytic arguments. As remarked in the introduction, it's known that every integer is the sum of four squares, and if n is a sum of two squares then if p is a prime congruent to 3 modulo 4 dividing n then the highest power of p dividing n is even (see for example [HW]). While proving this is beyond the scope of this note, we can easily prove the following.

Theorem 3.1. *If n_1 and n_2 are sums of two squares, so is $n_1 n_2$.*

Proof. The claim follows immediately from the multiplicativity of the complex norm. For $k \in \{1, 2\}$ let $n_k = x_k^2 + y_k^2$, and set $z_k = x_k + iy_k$, where $i = \sqrt{-1}$. The norm (or length) of z_k, denoted $|z_k|$, is $\sqrt{z_k \overline{z_k}}$, where $\overline{z_k} = x_k - iy_k$ is the complex conjugate of z_k. As the norm of a product is the product of the norms (with this also holding for the square of the lengths) we should have $|z_1 z_2| = |z_1||z_2|$. We prove this by brute force algebra. As $|z_k|^2 = x_k^2 + y_k^2$ (since $i^2 = -1$) we find

$$\begin{aligned} |z_1 z_2|^2 &= |(x_1 + iy_1)(x_2 + iy_2)|^2 \\ &= |(x_1 x_2 - y_1 y_2) + i(x_1 y_2 + x_2 y_1)|^2 \\ &= (x_1 x_2 - y_1 y_2)^2 + (x_1 y_2 + x_2 y_1)^2 \\ &= (x_1^2 x_2^2 - 2x_1 x_2 y_1 y_2 + y_1^2 y_2^2) + (x_1^2 y_2^2 + 2x_1 x_2 y_1 y_2 + x_2^2 y_1^2) \\ &= x_1^2 x_2^2 + x_1^2 y_2^2 + x_2^2 y_1^2 + y_1^2 y_2^2 \\ &= x_1^2(x_2^2 + y_2^2) + y_1^2(x_2^2 + y_2^2) \;=\; (x_1^2 + y_1^2)(x_2^2 + y_2^2) \;=\; |z_1|^2 |z_2|^2. \end{aligned} \tag{3.1}$$

Thus the multiplicativity of the norm (or length) of complex numbers shows that the product of two numbers that are sums of two squares is itself a sum of two squares. □

Remark 3.2. *There's a generalization of complex numbers, the quaternions, which have a similar multiplicative property and this plays a key role in proving all integers are the sum of four squares. We show every prime is the sum of at most four squares, and then use a similar multiplicativity argument to show if n_1 and n_2 are the sum of four squares so too is their product.*

3.1. *Upper Bound for Sums of Two Squares*

We now prove Theorem 1.2 with a similar argument as in Section 2.2. As

$$(x^2 + y^2) \bmod N \;=\; \big((x^2 \bmod N) + (y^2 \bmod N)\big) \bmod N, \tag{3.2}$$

we see the remainder modulo N of a sum is the remainder modulo N of the sums of the remainders modulo N. Thus it suffices to look at the sum of the remainders of the squares modulo N. We proceed as follows.

(1) Fix an integer N and for $x \in \{0, 1, 2, \ldots, N-1\}$ calculate x^2 mod N to obtain a set of remainders, $\mathcal{R}_N$. For example, if $N = 4$ then $\mathcal{R}_N = \{0, 1, 0, 1\}$.
(2) Remove any duplicate remainders from $\{\mathcal{R}_N\}$, keeping only the unique values to form a new set $\{\mathcal{U}_N\}$. When $N = 4$ we have $\{\mathcal{U}_N\} = \{0, 1\}$.
(3) For every pair of numbers from $\{\mathcal{U}_n\}$ (they can be the same, and without loss of generality we may assume the first element is at most the second), add them and take the result modulo N. Collect these sums into a final set, $\{\mathcal{S}_N\}$. For instance, if $N = 4$ there are three pairs (0,0), (0,1) and (1,1), and their sums modulo 4 give us the set $\{S_N\} = \{0, 1, 2\}$. Note in this case 25% of numbers are excluded as we cannot have a sum of two squares be 3 modulo 4.
(4) The cardinality of $\mathcal{S}_N$, denoted $|\mathcal{S}_N|$, gives an upper bound for the percentage of remainders that *have a chance of being expressible as the sum of two squares*: $\frac{|\mathcal{S}_N|}{N} \cdot 100\%$.

As we did in Section 2.2, we computed the record-minimum ratio percentages for the sum of two squares modulo N for N up to 10,000. The results are summarized below in Table 3.1.

N	Percentage	Factorization of N
1	100.00%	1
4	75.00%	2^2
8	62.50%	2^3
16	56.25%	2^4
32	53.13%	2^5
64	51.56%	2^6
72	48.61%	$2^3 \times 3^2$
144	43.75%	$2^4 \times 3^2$
288	41.32%	$2^5 \times 3^2$
576	40.10%	$2^6 \times 3^2$
1152	39.50%	$2^7 \times 3^2$
2304	39.19%	$2^8 \times 3^2$
7056	38.39%	$2^4 \times 3^2 \times 7^2$

Table 3.1. Values of N up to 10,000 with record-minimum percentages for the sum of two squares

Note $N = 4$ is already enough to show a positive percentage of numbers are not the sum of two squares, while $N = 72$ is the first modulus to prove that a majority of integers do not have such a representation. For N at most 10,000 we observe our lowest percentage, approximately 38.39%, at 7056. We will comment more extensively on patterns in the table in Section 4, and content ourselves for a few quick thoughts here. Every N with a percentage under 50% is composite and divisible by 2 to at least the third power, and If N has any other prime factors then they're of the form of $4k + 3$ and their exponents are always even numbers.

Brute-forcing the sum of squares requires a massive amount of computation. We ran our program on the same system as before: AMD Ryzen 7 5700U (1.80 GHz) with Radeon Graphics, 16.0 GB physical RAM (13.9 GB usable) and 64-bit x64 operating system; it took 811.7 seconds to compute up to 10,000 was 811.7s. When $N > 10^5$,

standard computers running simple code could not compute the results in a short time. However, noticing that these record-low N values share a highly specific prime factor structure (high powers of 2 and even exponents for $4k+3$ primes), we optimized our code using the DeepSeek LLM. For small values of N we checked the results of these algorithms with the simpler but slower straightforward code. We record the code used to investigate the sums of squares congruences in Section B.1. That code was fed into Deepseek, which modified the initial algorithm to the one in B.2, which resulted in significant gains in efficiency. All record-minimum N had the resulting value checked by the slower program.

3.2. *Comparison with Analytic Methods*

There are several approaches using analytic methods to bound the percentage of integers at most N that are the sum of two squares. Assume $x^2+y^2 \le N$ and without loss of generality we *could* take $x \le y$; we'll see the issue this creates, and fix shortly. Clearly $y \le \sqrt{N}$ (as otherwise the sum of squares exceeds N) and then x is at most $\min\left(\sqrt{N-y^2},\ y\right)$. Thus the number of such pairs is

$$\sum_{y=0}^{\sqrt{N}} \sum_{x=0}^{\min\left(\sqrt{N-y^2},\ y\right)} 1. \tag{3.3}$$

The presence of the minimum in the upper bound complicates the summation, so it's simpler to *not* assume $x \le y$. This means if $a^2+b^2 \le N$ we count it twice if $a \ne b$, as (a,b) and as (b,a). As the "diagonal" terms where $a=b$ are of size $\sqrt{N}$, which is a negligible amount of integers at most N, we can ignore this and instead drop the minimum, *remembering we are double counting and put in a factor of* $1/2$. Thus we are studying

$$\frac{1}{2}\sum_{y=0}^{\sqrt{N}} \sum_{x=0}^{\sqrt{N-y^2}} 1. \tag{3.4}$$

Unfortunately the square-root in the upper bound of the x-sum creates challenges; the sum is

$$\frac{1}{2}\sum_{y=0}^{\sqrt{N}} \lfloor \sqrt{N-y^2}+1 \rfloor. \tag{3.5}$$

Removing the floor function introduces a negligible error, but the sum over y is a challenge due to the summand being $\sqrt{N-y^2}$.

One option is to replace this sum with an integral; up to errors of size $\sqrt{N}$ (basically from dropping the floor function and the $+1$) we have the number of such pairs is

$$\frac{1}{2}\int_{y=0}^{\sqrt{N}} \sqrt{N-y^2}dy \ = \ \frac{1}{2}\int_{t=0}^{1} \sqrt{N(1-t^2)}\sqrt{N}dt \ = \ \frac{1}{2}N\int_{t=0}^{1} \sqrt{1-t^2}dt, \tag{3.6}$$

where we set $y = t\sqrt{N}$ so $dy = \sqrt{N}dt$. This gives, as $N \to \infty$, an upper bound on the

number of integers at most N that are the sum of two squares: if we assume all pairs give distinct elements (which of course is false, as for example $1^2 + 7^2 = 5^2 + 5^2$) then the maximum number of integers so represented is

$$\frac{\frac{1}{2}N\int_{t=0}^{1}\sqrt{1-t^2}dt}{N} \;=\; \frac{1}{2}\int_{t=0}^{1}\sqrt{1-t^2}dt, \tag{3.7}$$

which is clearly less than 1 (i.e., 100%) as the integrand is less than 1 for positive t. To find out how much less we need to evaluate this integral; the simplest is to do a trig substitution: if $t = \sin\theta$ then $dt = \cos\theta d\theta$ and

$$\frac{1}{2}\int_{t=0}^{1}\sqrt{1-t^2}dt \;=\; \frac{1}{2}\int_{\theta=0}^{\pi/2}\cos\theta\cdot\cos\theta d\theta \;=\; \frac{1}{2}\int_{\theta=0}^{\pi/2}\cos^2\theta d\theta. \tag{3.8}$$

While this integral can be evaluated using double angle identities, by symmetry it's one-fourth the integral of $\cos^2\theta$ from 0 to 2π, and this integral is the same as the integral of $\sin^2\theta$ (so it's one-half the sum of this and the sine-squared integral, which can be simplified as $\cos^2\theta + \sin^2\theta = 1$), thus

$$\begin{aligned}\frac{1}{2}\int_{t=0}^{1}\sqrt{1-t^2}dt &= \frac{1}{4}\frac{1}{2}\int_{\theta=0}^{2\pi}\cos^2 d\theta \\ &= \frac{1}{2}\frac{1}{4}\frac{1}{2}\int_{\theta=0}^{2\pi}\left[\cos^2\theta+\sin^2\theta\right]d\theta \\ &= \frac{1}{16}\int_{\theta=0}^{2\pi}1d\theta \;=\; \frac{2\pi}{16} \;=\; \frac{\pi}{8} \;\approx\; 39.3\%.\end{aligned} \tag{3.9}$$

Remark 3.3. *Notice that this is a worse bound than the congruence argument when the modulus is 7056, which gave about 38.39%, and is a lot more work! Further, we cannot push this analytic argument further. This is the best we can get; however, there is a chance of further improving the bound from the congruence argument through further search. Thus simple congruence arguments and a modest search give a better bound than we can get with our analytic argument and calculus.*

Remark 3.4. *We could get to the above analytic bound another way, as our count is the famous Gauss Circle Problem (see [BKZ, Iw]). For nice regions in the plane, the number of integer lattice points in the set equals the area with an error at most on the order of the length of the boundary (in many situations, such as the circle, one can obtain a power savings in the error term; however, such gains do not lead to any improvement in the limiting percentage bound).*[2] *In this case, we have a circle of radius $\sqrt{N}$, and the number of lattice points $0 \le x \le y \le \sqrt{N}$ would thus have main term of size $\frac{1}{8}\pi\sqrt{N}^2$ and an error of size at most order $\sqrt{N}$. Thus the main term is $N\pi/8$, exactly what we obtained before but* without *having to appeal to calculus.*

[2]To prove the claim with an error on the order of $\sqrt{N}$ note the number of lattice points is bounded below by the area of a wedge that is one-eighth of a circle with center at $(2,2)$ with radius $\sqrt{N}-4$ and at most that of a circle with center at $(-2,-2)$ and radius $\sqrt{N}+4$.

4. Discussion and Future Work

The tables of upper bounds arising from modular arguments suggest many intriguing conjectures. It's interesting that for Fibonacci numbers it's not until $N = 29$ that we can conclude a positive percentage of numbers are not the sum of two such elements. For sums of squares we reach this conclusion almost immediately, at $N = 4$. Thus, even though in the limit far more numbers are sums of two squares than sums of two Fibonaccis, the congruence arguments detect that a positive percentage cannot be done significantly earlier in the Fibonacci case than in the sum of squares case. However, as the moduli increase, the bounds for the Fibonacci case improve faster than those for the squares; by 377 the percentage is lower for the Fibonaccis.

We invite fellow math enthusiasts to join us in investigating these problems (email the first named author at sjm1@williams.edu). We also encourage people to visit the On-Line Encyclopedia of Integer Sequences, the OEIS [Sl]; this is a terrific resource to see connections, applications and properties of sequences, often with numerous references, explicit formulas, and connections with various problems.

4.1. *Conjectures on the Sum of two Fibonaccis*

For the sum of two Fibonacci numbers, the percentage plummets rapidly, falling below 50% by $N = 199$ and is around 3.34% by $N = 9349$. We hypothesize that as $N \to \infty$, congruence arguments with appropriate moduli will show this percentage converges to 0%. Below are several questions worth studying further, though some of these are known using more advanced methods.

- Prove there is a sequence of moduli N_k such that the upper bound on the percentage of numbers that are the sum of two Fibonaccis tend to zero.
- Explicitly construct such a sequence $\{N_k\}$.
- Determine properties (factorization, Pisano periods $\pi(N_k)$) of your sequence.
- Determine more terms in the record-minimum percentage sequence, $\{1, 29, 55, 76, 144, 199, 377, 521, 987, 1364, 2255, \dots\}$.
- Determine properties (factorization, Pisano periods) of the record-minimum percentage sequence.
- Generalize all these questions from the Fibonacci sequence to other, similar recurrence relations. One option is to keep the recurrence (each term is the sum of the previous two) but change the initial conditions; another possibility is to change the recurrence (perhaps keeping it second order, perhaps making it third or higher order).
- Ask the same question with other sequences, such as triangular numbers, pentagonal numbers, and so on.

4.2. *Conjectures on the Sum of two squares*

As remarked, relative to the case of the sum of two Fibonacci numbers the upper bound for the percentage of numbers that are the sum of two squares declines more gently. It inches down from 48.61% (at $N = 72$) to 38.39% (at $N = 7056$). Extending our calculation, we find a relatively minuscule drop of just over 4% down to 34.16% after increasing the range of moduli by almost a factor of 10, coming at $N = 903,168$. We can ask similar questions as in the previous subsection.

As we go further and further will the percentage converge to a non-zero limit, or

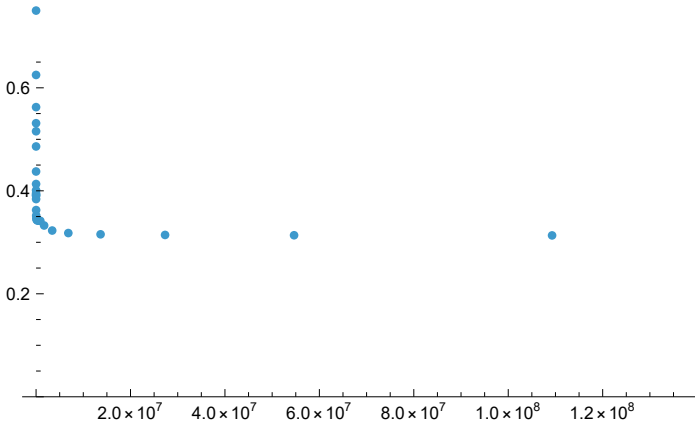

Figure 4.1. Plot of record-minimum percentages versus corresponding moduli.

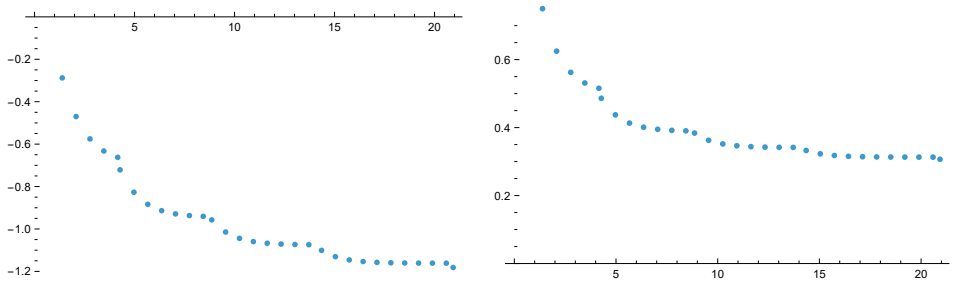

Figure 4.2. Plot of record-minimum percentages versus corresponding moduli (left is log-log, right is semi-log plot).

will it fall all the way to zero? For sequences with very slow decay, it can be hard to get enough data to clearly see; Figure 4.1 plots the record-low moduli and the corresponding percentages up to around a billion.

It can be hard to extract information from plots such as the above, as the x-axis spans so many orders of magnitude and it is hard to visually parse. A standard technique is to look at either a log-log plot (where we plot the logarithms of each component) or a semi-log plot (where we plot the percentages against the logarithm of the modulus). We show both plots in Figure 4.2.

The log-log and semi-log plots show an interesting structure. There data suggests that are curves where the percentage falls for awhile, and then there is a jump to a new curve of moduli where it falls again at a lower probability; however, it is not clear if the probability will reach zero or stabilize at some positive value.

Analyzing the data in more depth, the fact that all sub-50% ratios require at least 2^3 and even powers for prime factors of the form of $4k+3$ strongly implies that the structure of square remainders modulo N is entirely dictated by the power-of-two component and the number of $4k+3$ primes with even powers; in other words, the data suggests the curves or families we noticed above.

- Prove the congruence arguments yield a sequence of moduli $\{N_k\}$ where the resulting upper bounds converge from above to some constant c; for example, consider the bounds coming from $N_k = 2^k$.

- Building on the previous question, what is the smallest c where you can find an explicit sequence N_k whose upper bounds converge to c? In particular, is the smallest c attainable by these congruence methods zero or positive? (Note the analytic methods could not get below $\pi/8$, even though it is a theorem that the percentage is zero.)
- Determine more terms in the record-minimum percentage sequence, $\{1, 4, 8, 16, 32, 64, 72, 144, 288, 576, 1152, 2304, 7056, \dots\}$.
- Determine properties of the record-minimum percentage sequence. In particular, after the first term does every term have at least one factor of 2 in its factorization (i.e., is even)? For each $j \geq 1$ is there some modulus such that for all record-minimum percentages from that modulus and onward they are divisible by 2 at least j times? Once a prime is a factor of a record-minimum modulus is it a factor of all subsequent ones?

We did some preliminary investigations related to the above questions using AI (Claude and Aristotle). We inputted our record-minimum N and precentages and asked AI to find formulas two ways: telling it where these numbers came from, and just giving the sequence without context. Both systems gave terrific results that can be used as a great starting point to solve the conjectures. They identified that the answer splits as factors of what happens at prime powers, that there is no loss from primes congruent to 1 modulo 4, that there is a simple formula for the prime 2, and that there is a decrease coming every time our N is divisible by at least a square of a prime 3 mod 4. In this case the AI's suggested and then proved what the formula is. Interestingly, our initial program accidentally rounded the numerators in the probabilities for the record-minimum moduli above two billion; the AI's suggested that there was an issue with the data, asked if it had been rounded, and gave and proved a formula that described the rest of the data.

Additionally, both programs identified good sources in the literature for more information on congruence arguments; we are deliberately not giving the formulas and references so as not to ruin the fun for others who wish to study these problems. We can turn these suggestions into congruence argument proofs that the percentage of numbers representable as a sum of two squares tends to zero; however to do so the argument is no longer entirely elementary. Key ingredients include the Chinese Remainder Theorem (which is not too hard to prove), and more importantly the fact that the sum of the reciprocals of the primes congruent to 3 mod 4 diverges. We elect not to pursue such arguments here as our goal is a self-contained presentation that can be easily incorporated in an Algebra I class, and as stated content ourselves with remarking that we have gathered enough data that AI can suggest formulas which can be proved which, with additional analytic facts, resolve the original question: the limiting percentage is zero.

Acknowledgement and Statement of AI Assistance

The first named author expresses his thanks to Mr. Li Huang for his valuable guidance, constructive suggestions, and generous assistance during the writing and revision of this paper. Both authors thank the Math League (see https://mathleague.com/) where they met when the second named author was lecturing.

An initial draft of some subsections in Chinese was translated and polished using AI, which then converted that text into LaTeX source code, which was then extensively

edited by the authors. AI was also used to replace the original C++ code to investigate the remainders of sums of squares modulo N with a more efficient algorithm, allowing us to test a significantly larger range of moduli; in addition to checking the new code line by line, we also tested its output against our slower, simpler program and found perfect agreement. All AI-assisted content has been carefully reviewed, verified, and finalized by the authors. The core ideas, research design, analysis, and conclusions of this work were independently completed by the authors; AI was only used for the purposes described above.

Appendix A. C++ Code for Calculating the Remainder Coverage of the Sum of Two Fibonacci Numbers

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int n, a, b, cnt, m;
    float minn = 1;
    cin >> m;
    for (n = 2; n < m; n++) {
        a = 0, b = 1, cnt = 0;
        vector<bool> vis(n * n);
        bool v[n + 1], visit[n + 1];
        memset(v, 0, sizeof(v));
        memset(visit, 0, sizeof(visit)); // initialization
        vector<int> ans;
        ans.push_back(0);
        visit[0] = 1;
        while (!vis[a * n + b]) { // compute the period of remainders
            if (!visit[b]) ans.push_back(b); // store new remainder
            visit[b] = 1;
            vis[a * n + b] = 1;
            int c = (a + b) % n;
            a = b; b = c;
        }
        for (int i = 0; i < ans.size(); i++) { // enumerate the
recorded remainders
            for (int j = 0; j < ans.size(); j++) {
                if (!v[(ans[i] + ans[j]) % n]) cnt++; // count
distinct sums modulo n
                v[(ans[i] + ans[j]) % n] = 1;
            }
        }
        if ((float)cnt / n < minn) { // calculate probability
            minn = (float)cnt / n;
            cout << n << " " << cnt << " " << minn << endl;
        }
        else if (cnt != n) {
            cout << n << " " << cnt << " " << (float)cnt / n << endl;
```

```
// output final result
        }
    }
    return 0;
}
```

Appendix B. C++ Code for Calculating the Remainder Coverage of the Sum of Two Squares

B.1. *The Original C++ Code Before DeepSeek Optimization*

```
#include <bits/stdc++.h>
using namespace std;

int main() {
    int l, r, ans = 0;
    float minn = 1;
    cin >> r;
    bool v[r + 1];
    for (int i = 1; i <= r; i++) {
        float s = 0;
        memset(v, 0, sizeof(v));
        for (int j = 1; j <= i; j++) // start enumeration
            for (int k = 1; k <= j; k++) {
                int tmp = (j * j + k * k) % i; // enumerate remainders
of (sum of squares) modulo i for all natural numbers less than i
                if (!v[tmp]) { // check if duplicate; if not, record
it
                    v[tmp] = 1;
                    s++; // count number of distinct remainders
                }
            }
        if (minn > s / i) { // calculate probability
            minn = s / i;
            ans = i;
        }
    }

    cout << ans << " " << fixed << minn * 100 << "%" <<
setprecision(6) << endl;

     return 0;
}
```

B.2. *The Upgraded C++ Code After DeepSeek Optimization (Fix Rounding Error)*

```
#include <iostream>
#include <vector>
```

```
using namespace std;

int main() {
    int r;
    cin >> r;
    if (r < 1) return 0;
// smallest prime factor sieve for fast factorization
    vector<int> lp(r + 1, 0);
    vector<int> primes;
    for (int i = 2; i <= r; ++i) {//linear prime sieve
        if (lp[i] == 0) {
            lp[i] = i;
            primes.push_back(i);
        }
        for (int p=0;p<primes.size();p++) {
            if (primes[p] > lp[i] || i * primes[p] > r) break;
            lp[i * primes[p]] = primes[p];
        }
    }
// ratio[i] = maximum proportion of integers representable as sum of
two squares modulo i
    vector<double> ratio(r + 1, 1.0);
    vector<int> cnt(r+1,1);
ratio[1] = 1.0;
// compute coverage ratio for each modulus using prime-power
decomposition rules
    for (int i = 2; i <= r; ++i) {//calculate the ratio
        int temp = i;
        double prod = 1.0;
        while (temp > 1) {
            int p = lp[temp];
            int e = 0;
            long long p_pow = 1; // count exponent e and compute p^e
            while (temp % p == 0) {
                temp /= p;
                e++;
                p_pow *= p;
            }
            long long f;
            if (p == 2) { // rule for prime factor 2
                if (e == 1) f = 2;
                else f = (1LL << (e - 1)) + 1; // 2^{e-1} + 1
            } else {
                if (e == 1) {
                    f = p;
                } else {
                    if (p % 4 == 1) {// rule for primes congruent to 1
mod 4
                        f = p_pow; // p^e
                    } else { // rule for primes congruent to 3 mod 4
```

```
                        // f(p,e) = p^e - p^{e-1} + p^{e-2}
                        long long p_pow_e_minus_1 = p_pow / p;
                        long long p_pow_e_minus_2 = p_pow_e_minus_1 /
p;
                        f = p_pow - p_pow_e_minus_1 + p_pow_e_minus_2;
                    }
                }
            }
            prod *= (double)f / p_pow;
            cnt[i]*=f;
        }
        ratio[i] = prod;
    }
// find modulus with minimal representable proportion
    double min_ratio = 1.0;
    int ans = 1;
    for (int i = 1; i <= r; ++i) {//find the minimum ratio
        if (ratio[i] < min_ratio) {
            min_ratio = ratio[i];
            ans = i;
        }
    }

    cout << "Record-Minimum N " << ans << endl;
    printf("Percentage %.8f%%\n", min_ratio * 100);

    return 0;
}
```

References

[BKZ] B. C. Berndt, S. Kim and A. Zaharescu, *The Circle Problem of Gauss and the Divisor Problem of Dirichlet—Still Unsolved*, The American Mathematical Monthly **125** (2018), No. 2 99–114. https://www.jstor.org/stable/48661825.

[Fe] P. de Fermat, *Margin note in Diophantus' Arithmetica*, 1637.

[Go] C. Goldbach, *Letter to Leonhard Euler*, 1742.

[HW] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers*, Oxford University Press, 1979.

[He] H. A. Helfgott, *The ternary Goldbach conjecture is true*, arXiv preprint (2015, version 2): https://arxiv.org/abs/1501.05438.

[Iw] H. Iwaniec, *Introduction to the Spectral Theory of Automorphic Forms*, Revista Matemática Iberoamericana, 1995

[KKMW] M. Kologlu, G.S. Kopp, S. J. Miller, Y. Wang, *On the Number of Summands in Zeckendorf Decompositions, Fibonacci Quarterly.* **49** (2011), no. 2, 116–130.

[Ko] T. Koshy, *Fibonacci and Lucas Numbers with Applications*, Wiley-Interscience, New York, 2001

[Na] M. Nathanson, *Additive Number Theory: The Classical Bases*, Graduate Texts in Mathematics, Springer-Verlag, New York, 1996.

[MT-B] S. J. Miller and R. Takloo-Bighash, *An Invitation to Modern Number Theory*, Princeton University Press, Princeton, NJ, 2006.

[Pa] J. Pavlus, *Sum-of-Three-Cubes Problem Solved for 'Stubborn' Number 33*, Quantum Magazine, March 26, 2019. https://www.quantamagazine.org/sum-of-three-cubes-problem-solved-for-stubborn-number-33-20190326/.

[Sl] N. J. A. Sloane (founder), the On-Line Encyclopedia of Integer Sequences (OEIS), https://oeis.org.

[SS] E. Stein and R. Shakarchi, *Complex Analysis*, Princeton University Press, Princeton, NJ, 2003.

[TW] R. Taylor and A. Wiles, *Ring-theoretic properties of certain Hecke algebras*, Ann. Math. **141** (1995), 553–572.

[Vi] I. M. Vinogradov, *Representation of an odd number as a sum of three primes*, *Doklady Akademii Nauk SSSR* **15** (1937), 291–294.

[We1] E. W. Weisstein, *Chinese Remainder Theorem*, From MathWorld—A Wolfram Web Resource, 2026. https://mathworld.wolfram.com/ChineseRemainderTheorem.html

[We2] E. W. Weisstein, *Sum of Squares Function*, From MathWorld—A Wolfram Web Resource, 2026. https://mathworld.wolfram.com/SumofSquaresFunction.html

[Wi] A. Wiles, *Modular elliptic curves and Fermat's Last Theorem*, *Annals of Mathematics* **141**(3) (1995), 443–551.

[Wr] E. M. Wright, *An easier form of Waring's problem*, *Journal of the London Mathematical Society* **1**(4) (1934), 267–272.

[Ze] E. Zeckendorf, *Représentation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas, Bull. Soc. Roy. Sci. Liège.* **41** (1972), 179–182.