Texas A&M University

Research Project 1

Some Conjectures on Fibonacci Numbers

Authors:
Daniel Whatley
Sarah Sahibzada
Taylor Wilson

Supervisor:
Dr. Sara Pollock

October 21, 2015

Contents

1 Introduction

The *Fibonacci sequence*, $\{F_n\}$, is defined as follows:

$$F_1 = 1, \qquad F_2 = 1, \qquad F_n = F_{n-1} + F_{n-2} \quad \text{for} \quad n \geq 3.$$

In this paper we analyze two conjectures: one about which Fibonacci numbers are perfect squares, and one about factorials (put reference and more info here).

2 Theoretical Analysis

It was a long-standing conjecture, until it was proved in 1964 [2], that the only Fibonacci numbers that are perfect squares are $F_1 = 1$, $F_2 = 1$, and $F_12 = 144$. M. Wunderlich[1] described an ingenious "exclusion" method to calculate which Fibonacci numbers, out of the first million, can possibly be perfect squares. The method does not prove that any Fibonacci numbers are perfect squares, rather it rules out those that are not.

We also created a data structure to check Lucas number and factorial conjecture... (insert more detail) We also did identities for Lucas numbers...

3 Computational Approach

3.1 Some subsection thingy

[1] found that the only Fibonacci numbers up to $F_{1000000}$ are the three described above using a computational approach. The researchers here took a similar approach, but used optimization techniques to decrease running time and to give more conclusive results. The programming language used in this research was predominantly C++.

Consider a prime p. We first calculate the period of the Fibonacci numbers $\pmod p$, denoted here as the *Fibonacci period* of p. For example, the Fibonacci period of $p = 7$ is 16, as the Fibonacci sequence $\pmod 7$ repeats after 16 steps. Specifically, the first 16 Fibonacci numbers $\pmod 7$ are:

$$1, 1, 2, 3, 5, 1, 6, 0, 6, 6, 5, 4, 2, 6, 1, 0,$$

after which it repeats $1, 1, 2, 3, 5, \ldots$ again. Because the numbers $\pmod 7$ that are quadratic non-residues are 3, 5, and 6, replacing each quadratic residue by a 1 and each quadratic non-residue by a 0 gives the binary string

$$111001010001101.$$

Let this string for each prime p be S_p.

Suppose we take a list of primes: $p_1, p_2, \ldots, p_n$, and let the Fibonacci periods of each prime be $d_1, d_2, \ldots, d_n$. Take those indices of S_{p_i} that are 0. If any Fibonacci number is congruent to one of these indices mod d_i, then it can never be a perfect square of any integer. Such Fibonacci numbers can now be eliminated due to this prime. Repeat for more primes, until the number of possible square Fibonacci numbers is down to a reasonable number to analyze. We expect that approximately half of the Fibonacci numbers will be eliminated at each step, as about half of the integers mod a prime p are quadratic residues. This makes the overall running time $O(\log n)$, where n is the number of Fibonacci numbers considered.

The computation in [1] was somewhat limited for two reasons. First, for each prime p, the quadratic residues and non-residues mod p had to be re-iterated all the way to the computation limit (in the paper, the computation limit is 10^6). This means that a loop of size 10^6 had to be performed each time a new prime was considered. Second, because 10^6 is not perfectly divisible by each of the Fibonacci periods considered, the computation cannot easily be extended to all positive integers.

We now describe a method to optimize the approach.

4 Results

5 Conclusion

6 Individual Contributions

1. Sarah:
2. Daniel: I wrote a significant portion of the code for
3. Taylor:

7 References

1. Wunderlich paper (MLA citation here)
2. Paper in which squares conjecture proved
3. Other stuff about other conjectures here