by using increments and limits instead of differentials, but the present form helps to demonstrate the utility of the differential calculus in addressing such problems. This alternate approach will hopefully provide a better link between thought processes in both calculus and physics.

Reference

1. G. B. Thomas and R. L. Finney, *Calculus*, 9th ed., Addison-Wesley, 1996.

A Magic Trick from Fibonacci

James Smoak (jsmoak@worldnet.att.net), 12140 E. Iowa Drive, Aurora, CO 80012
Thomas J. Osler (Osler@rowan.edu), Mathematics Department, Rowan University, Glassboro, NJ 08028

A mathematical magician displays the following remarkable fractions and asks us to think “Fibonacci.”

$$\frac{100}{89} = 1.\ 1\ 2\ 3\ 5\ 955056\cdots$$

$$\frac{10000}{9899} = 1.\ 01\ 02\ 03\ 05\ 08\ 13\ 21\ 34\ 55\ 9046368\cdots$$

$$\frac{1000000}{998999} = 1.\ 001\ 002\ 003\ 005\ 008\ 013\ 021\ 034\ 055\ 089$$
$$144\ 233\ 377\ 610\ 98859958818777596\cdots$$

The decimal expansion of our first fraction generates the first five Fibonacci numbers before blurring into other digits. Our second fraction generates the first ten, and the third fraction generates the first fifteen. Notice that the successive fractions change by appending two 0s to the numerator and a 9 to the front and back of the denominator. Will the next fraction $\frac{100000000}{99989999}$ generate the first twenty Fibonacci numbers in a similar way? Does this pattern continue forever? What is behind this magic trick?

A check of several additional fractions with a computing aide like Mathematica shows that the pattern does appear to continue. (Roberts [**2**] mentions the fraction 10000/9899.)

We now reveal the machinery of the magician. We use the familiar notation $F_1 = 1$, $F_2 = 1$, $F_3 = 2, \ldots$ for the Fibonacci numbers, with the recurrence relation $F_n = F_{n-1} + F_{n-2}$. The generating function [**1**] for these numbers is the key item

$$\frac{1}{1 - x - x^2} = F_1 + F_2 x + F_3 x^2 + F_4 x^3 + \cdots. \tag{1}$$

Note that

$$(F_1 + F_2 x + F_3 x^2 + \cdots)(1 - x - x^2) = F_1 + x(F_2 - F_1) + \sum_{n=2}^{\infty} x^n (F_{n+1} - F_n - F_{n-1}) = 1$$

since $F_1 = 1$, $F_2 - F_1 = 0$, and $F_{n+1} - F_n - F_{n-1} = 0$, for all $n \geq 2$.

Now letting $x = 0.1$, we get

$$\frac{1}{1-0.1-0.01} = \frac{100}{89} = 1 + 1(.1) + 2(.01) + 3(.001) + 5(.0001) + 8(.00001) + 13(.000001) + \cdots.$$

This explains how the first fraction displays the first five Fibonacci numbers. The sixth Fibonacci number, 8, becomes 9 in the decimal expansion because of carry from the number 13.

In the same way, let $x = 0.01$ in (1) and get

$$\frac{1}{1-0.01-0.0001} = \frac{10000}{9899} = 1 + 1(.01) + 2(.0001) + 3(.000001) + 5(.00000001) + \cdots.$$

This explains how the second fraction displays the first ten Fibonacci numbers using two digits per number.

Letting $x = 0.001$ in (1) generates our third number $\frac{1000000}{998999}$ in a similar way.

In general, we are generating fractions of the form $\frac{10^{2m}}{10^{2m}-10^m-1}$, with $m = 1, 2, 3, \ldots$. In its decimal expansion, we display each consecutive Fibonacci number F_n using a block of m digits. This will work well until the next Fibonacci number has $m+1$ digits, causing carry and destruction of the pattern.

Another explanation uses only long division. Study the following calculation of $\frac{10^{2m}F_1}{10^{2m}-10^m-1}$:

$$\begin{array}{r|l}
 & F_1 + 10^{-m}F_2 + 10^{-2m}F_3 + 10^{-3m}F_4 + \cdots \\
10^{2m} - 10^m - 1 & 10^{2m}F_1 \\
 & 10^{2m}F_1 - 10^m F_1 - F_1 \\
 & \quad 10^m F_2 + F_1 \\
 & \quad 10^m F_2 - F_2 - 10^{-m}F_2 \\
 & \qquad F_3 + 10^{-m}F_2 \\
 & \qquad F_3 - 10^{-m}F_3 - 10^{-2m}F_3 \\
 & \qquad\quad 10^{-m}F_4 + 10^{-2m}F_3 \\
 & \qquad\quad 10^{-m}F_4 - 10^{-2m}F_4 - 10^{-3m}F_4 \\
 & \qquad\qquad 10^{-2m}F_5 + 10^{-3m}F_4
\end{array}$$

Since only m spaces are allowed for the digits, the pattern terminates with the first F_k that has $m+1$ digits.

Usually there is a run of five consecutive Fibonacci numbers with the same number of digits, but a computer search shows that about 5% of the time, there are only four Fibonacci numbers in the run. Why is this so? A rough calculation can be made using Binet's formula [**1**], $F_n = \frac{\phi^n - (-\phi)^{-n}}{\sqrt{5}}$, with $\phi = \frac{1+\sqrt{5}}{2} = 1.61803\ldots$. This enables us to calculate F_n directly. Since $(-\phi)^{-n}$ is small, we can calculate the Fibonacci numbers exactly by simply rounding off the expression $F_n \approx \frac{\phi^n}{\sqrt{5}}$. How many digits does F_n have? Let $d(n)$ denote the number of decimal digits in F_n. Suppose we write $F_n = d_N d_{N-1} \cdots d_2 d_1$, where each d_k is a decimal digit. Now $F_n = 10^{N-1} d_N . d_{N-1} \cdots d_1$ and $\log_{10} F_n = N - 1 + \log_{10} d_N . d_{N-1} \cdots d_1 =$

$N-1+y$, where $0 \le y < 1$. Thus, $d(n) = N = 1 + \log_{10} F_n - y = \lfloor 1 + \log_{10} F_n \rfloor$, where $\lfloor z \rfloor$ denotes the greatest integer less than or equal to z. Using the approximation $F_n \approx \frac{\phi^n}{\sqrt{5}}$ we get $d(n) = \lfloor 1 + \log_{10}(\frac{\phi^n}{\sqrt{5}}) \rfloor \approx \lfloor 1 + n \log_{10} \phi - \log_{10} \sqrt{5} \rfloor \approx \lfloor 0.209n + 0.651 \rfloor$. Notice that $d(n+k) - d(n) \approx \lfloor 0.209n + 0.651 + 0.209k \rfloor - \lfloor 0.209n + 0.651 \rfloor$. Notice that for $k = 1, 2, 3$ or 4, $d(n+k) - d(n) = 0$ or 1. But $d(n+5) - d(n) = 1$ or 2. When $d(n+5) - d(n) = 1$, we get a run of five Fibonacci numbers with the same number of digits. However, when $d(n+5) - d(n) = 2$, we get only four. Using a computer, we found that $d(n+5) - d(n) = 2$ at $n = 16, 35, 59, 83, 102, 126, 150, 169, 193, \ldots$. For example, F_{16} to F_{21} are 987, 1597, 2584, 4181, 6765, 10946, which shows a run of only four Fibonacci numbers with four digits.

Acknowledgment. The authors wish to thank George Andrews for his support and encouragement.

References

1. R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics*, Addison-Wesley, 1989, pp. 283–286.
2. Joe Roberts, *Elementary Number Theory*, MIT Press, 1977, p. 16.

Lagrange Multipliers Can Fail To Determine Extrema

Jeffrey Nunemacher (jlnunema@cc.owu.edu), Ohio Wesleyan University, Delaware, OH 43015

The method of Lagrange multipliers is the usual approach taught in multivariable calculus courses for locating the extrema of a function of several variables subject to one or more constraints. It must, however, be applied with care because the method can miss the sought extremal values. This capsule discusses some simple examples in which Lagrange multipliers fails to locate extrema.

Recall that the method of Lagrange multipliers proceeds as follows in the simplest two dimensional setting. To find the extrema of a function $f(x, y)$ subject to the constraint $g(x, y) = k$ when all functions are C^1 smooth, we compute the gradient vectors $\nabla f(x, y)$ and $\nabla g(x, y)$ and solve the simultaneous system in three variables x, y, and λ

$$\nabla f(x, y) = \lambda \nabla g(x, y) \quad \text{and} \quad g(x, y) = k. \tag{1}$$

Then if the geometry is right, a constrained extremum must occur at a point (x_0, y_0) among the solutions to (1). Since this set is often finite, the location of the extrema can be determined by surveying all possibilities. But to be assured that the method succeeds, we must know that the geometry is right—that is, the set defined by $g(x, y) = k$ is a smooth curve in the plane. Here the Implicit Function Theorem is useful; it guarantees that a level set $g(x, y) = k$ is a smooth curve with nonvanishing tangent vector in a neighborhood of a point (a, b) if $\nabla g(a, b) \neq \mathbf{0}$. Thus, when seeking constrained extrema, we should also examine all critical points of $g(x, y)$.

