refers to a sequence of the fourth level, whose terms consist of ordered s-tuples of consecutive terms of $F^{(q,r)}$; this $F^{(q,r)}$ in its turn is a sequence of the third level, whose terms consist of ordered r-tuples of consecutive terms of $F^{(q)}$; and $F^{(q)}$ is a sequence of the second level whose terms consist of ordered q-tuples of consecutive terms of f.

The hierarchy results in a generalization of Theorem 1.

Theorem 2: Let $f\colon Z \to Z$ with $f_0 = 0$, $f_1 = 1$, and $f_n = f_{n-2} + f_{n-1}$. For every $m \in N$, let $q_1, \ldots, q_m \in N - \{1\}$. Let

$$Z^{(q_1)} = \{(f_{n-1}, f_n, \ldots, f_{n+q_1-2}) : n \in Z\},$$

the set of all ordered q_1-tuples of consecutive terms of f. Let $F^{(q_1)}\colon Z \to Z^{(q_1)}$ with

$$F^{(q_1)} = (f_{n-1}, f_n, \ldots, f_{n+q_1-2}).$$

Further, let

$$Z^{(q_1, \ldots, q_m)} = \left\{\left(F_{n-1}^{(q_1, \ldots, q_{m-1})}, F_n^{(q_1, \ldots, q_{m-1})}, \ldots, F_{n+q_m-2}^{(q_1, \ldots, q_{m-1})}\right) : n \in Z\right\},$$

the set of all ordered q_m-tuples of consecutive terms of $F^{(q_1, \ldots, q_{m-1})}$. Let

$$F_n^{(q_1, \ldots, q_m)}\colon Z \to Z^{(q_1, \ldots, q_m)}$$

with

$$F_n^{(q_1, \ldots, q_m)} = \left(F_{n-1}^{(q_1, \ldots, q_{m-1})}, F_n^{(q_1, \ldots, q_{m-1})}, \ldots, F_{n+q_m-2}^{(q_1, \ldots, q_{m-1})}\right).$$

Then $F^{(q_1, \ldots, q_m)}$ constitutes a two-sided sequence with terms $F_n^{(q_1, \ldots, q_m)}$, $n \in Z$, and the property

$$F_n^{(q_1, \ldots, q_m)} = F_{n-2}^{(q_1, \ldots, q_m)} + F_{n-1}^{(q_1, \ldots, q_m)}.$$

Moreover, the terms of $F^{(q_1, \ldots, q_m)}$ form an abelian group under the multiplication

$$F_n^{(q_1, \ldots, q_m)} F_p^{(q_1, \ldots, q_m)} = F_{n+p}^{(q_1, \ldots, q_m)}.$$

*****

EXPLORING AN ALGORITHM

DMITRI THORO and HUGH EDGAR
San Jose State University, San Jose CA 95192

Dedicated to the memory of our dear friend and colleague, Vern

1. INTRODUCTION

We start with a simple algorithm for generating pairs L (left column) and R (right column) of Fibonacci numbers. In a slightly modified version we wish to investigate the ratios L/R as the number of iterations $n \to \infty$. This, it turns out, involves (ancient) history, geometry, number theory, linear algebra, numerical analysis, etc.!

2. THE BASIC ALGORITHMS

Let us consider a "computer project" (appropriate for the first assignment in an Introduction to Programming course):

Given a suitable positive integer N, write a program which generates the *sum* of the first N Fibonacci numbers with *even* subscripts.

Of course one can generate F_{2i} and form a cumulative sum. A more imaginative student, however, might use the following algorithm.

Algorithm I:

(a) Input N
(b) $L \leftarrow 1,\ R \leftarrow 0$
(c) $L \leftarrow L + R,\ R \leftarrow L + R,\ N \leftarrow N - 1$
(d) If $N \neq 0$, go to step (c); else output $L + R - 1$ and stop.
["$\leftarrow$" means "is replaced by."]

Thus in BASIC PLUS we would write

```
10    INPUT N
20    L = 1 \     R = 0
30    L = L + R \     R = L + R \     N = N - 1
40    IF N<>0 THEN 30 ELSE PRINT A + B - 1
999   END
```

Or, on the TI 59 Programmable Calculator, we could *enter* N, *press* A, and execute:

```
LBL A   STO 00  1 STO 01  0 STO 02
LBL B   RCL 02  SUM 01  RCL 01  SUM 02
        DSZ 0 B RCL 01 + RCL 02 - 1 = R/S
```

[Here LBL, STO, RCL, SUM, DSZ, and R/S are codes for label, store, recall, sum, decrement-and-skip-on-zero, and run-stop, respectively. In particular, N is placed in memory location 00 and, after each pair of consecutive Fibonacci numbers is generated, the contents of loc. 00 is decreased by 1; if the result $\neq$ 0, we repeat by going back to "LBL B".]

The reader is invited to guess (or determine) the values of N for which our output doesn't exceed the 10 digits which are displayed on the TI 59.

If we started with $L = R = 1$, then the pairs L,R would have ratios L/R approaching the *golden mean* $(1 + \sqrt{5})/2$. Given a pair L,R let us now generate the next pair by slightly modifying the preceding algorithm.

Algorithm II: Given N $(N > 0)$

(a) $L \leftarrow 1,\ R \leftarrow 1$
(b) $T \leftarrow L + NR,\ R \leftarrow L + R,\ L \leftarrow T$
(c) Repeat step (b) if desired; else output L/R.

We wish to investigate the ratios L/R as the number of iterations $n \to \infty$.

3. PRELIMINARY OBSERVATIONS

Algorithm II can be described by the equations

$$\begin{cases} L_{k+1} = L_k + NR_k \\ R_{k+1} = L_k + R_k,\quad k = 0, 1, 2, \ldots, \end{cases}$$

where L_0, R_0, and $N > 0$ are given real numbers. We will use the matrix form

$$\begin{pmatrix} L_{k+1} \\ R_{k+1} \end{pmatrix} = A\begin{pmatrix} L_k \\ R_k \end{pmatrix} \text{ where } A = \begin{pmatrix} 1 & N \\ 1 & 1 \end{pmatrix}.$$

Two examples are:

$L_0 = R_0 = 1,\ N = 2$		$L_0 = R_0 = 1,\ N = 3$	
	(2)		(3)
1	1	1	1
3	2	4	2
7	5	10	6
41	29	28	16
⋮	⋮	⋮	⋮

The ratios L/R in each row are, indeed, the *convergents* of the continued fraction expansions of $\sqrt{2}$ and $\sqrt{3}$, respectively. (The reader is invited to try $L_0 = R_0 = 1$, $N = 7$.) Will this ever happen again?

4. AN ATTEMPT TO ACCELERATE CONVERGENCE

After consideration of additional examples, it becomes evident that for large N the values $L/R \to \sqrt{N}$ *slowly*. This suggests that we might be able to accelerate convergence by applying the algorithm to $1/N$ and then taking the reciprocal of the final approximation.

Unfortunately, this doesn't help. E.g., for $L_0 = R_0 = 1$, $N = 5$, we get ratios 1, 3, 2, 7/3, 11/5, 9/4, ..., while $N = 1/5$ yields ratios 1, 3/5, 1/2, 7/15, 5/11, 9/20, ...

In general,

$$R_{2k}(1/N) = 1/R_{2k}(N) \quad \text{and} \quad R_{2k+1}(1/N) = R_{2k+1}(N)/N.$$

Thus, if $R_i(N) \to \sqrt{N}$, then

$$R_{2k}(1/N) \to 1/\sqrt{N} \quad \text{and} \quad R_{2k+1}(1/N) \to \sqrt{N}/N = 1/\sqrt{N};$$

i.e., $R_i(1/N) \to 1/\sqrt{N}$ as $i \to \infty$ and, moreover, convergence is "at the same rate."

As we will later see, it is the size of the ratio

$$|g(N)| = \left|\frac{1 - \sqrt{N}}{1 + \sqrt{N}}\right|$$

which determines the rate of convergence; the smaller the ratio, the faster the convergence! Since $|g(1/T)| = |g(T)|$ the above idea is fruitless. Put another way, the closer N is to 1, rather than 0, the faster the convergence.

5. A MATRIX PROOF

(a) We start with the characteristic polynomial

$$f(\lambda) = \det(A - \lambda I) = \lambda^2 - 2\lambda + 1 - N$$

associated with the matrix

$$A = \begin{pmatrix} 1 & N \\ 1 & 1 \end{pmatrix}.$$

Solving $f(\lambda) = 0$ we get eigenvalues $\lambda_1 = 1 + \sqrt{N}$ and $\lambda_2 = 1 - \sqrt{N}$.

(b) Applying the division algorithm to the polynomials λ^k and $f(\lambda)$ (in the Euclidean domain $R[\lambda]$ of polynomials with real coefficients) yields

$$\lambda^k = (\lambda^2 - 2\lambda + 1 - N)g(\lambda) + r(\lambda)$$

where $r(\lambda) = \beta_1 + \beta_2\lambda$.

(c) Setting $\lambda = \lambda_1, \lambda_2$ we get

$$\lambda_1^k = \beta_1 + \beta_2\lambda_1 \quad \text{and} \quad \lambda_2^k = \beta_1 + \beta_2\lambda_2.$$

When solved simultaneously, one finds

$$\beta_1 = (\lambda_2\lambda_1^k - \lambda_1\lambda_2^k)/(\lambda_2 - \lambda_1) \quad \text{and} \quad \beta_2 = (\lambda_2^k - \lambda_1^k)/(\lambda_2 - \lambda_1).$$

(d) Invoking the Cayley-Hamilton theorem produces $A^k = \beta_1 I + \beta_2 A$ (where I is, as usual, the 2×2 identity matrix).

(e) Our original matrix equation can easily be written in the form

$$\begin{pmatrix} L_k \\ R_k \end{pmatrix} = A^k \begin{pmatrix} L_0 \\ R_0 \end{pmatrix}, \; k = 1, 2, 3, \ldots .$$

Using (c), (d), and a little algebra, we get

$$\begin{pmatrix} L_k \\ R_k \end{pmatrix} = \begin{pmatrix} (\beta_1 + \beta_2)L_0 + \beta_2 NR_0 \\ \beta_2 L_0 + (\beta_1 + \beta_2)R_0 \end{pmatrix}$$

or

$$\frac{L_k}{R_k} = \frac{\left(\dfrac{\beta_1}{\beta_2} + 1\right)L_0 + NR_0}{L_0 + \left(\dfrac{\beta_1}{\beta_2} + 1\right)R_0}$$

However, $\beta_1/\beta_2 = \dfrac{\lambda_2 - \lambda_1(\lambda_2/\lambda_1)^k}{(\lambda_2/\lambda_1)^k - 1} \to -\lambda_2$ as $k \to \infty$ (since $|\lambda_1| > |\lambda_2|$). Thus

$$\frac{L_k}{R_k} \to \frac{\sqrt{N}L_0 + NR_0}{L_0 + \sqrt{N}R_0} = \sqrt{N} \text{ as } k \to \infty.$$

6. SOME ACCIDENTS

Consider $Q \equiv \dfrac{\sqrt{N}L + NR}{L + \sqrt{N}R}$.

(a) Illegal Cancellation 1.1: "Erasing" the first term in the numerator and denominator of Q yields $Q = NR/(\sqrt{N}R) = \sqrt{N}$.

(b) Illegal Cancellation 1.2: "Erasing" the second term in the numerator and denominator yields $Q = \sqrt{N}L/L = \sqrt{N}$.

(c) Illegal Simplification 1.3: Setting $L = R = 1$, we get

$$Q = (\sqrt{N} + N)/(1 + \sqrt{N}) = \sqrt{N}.$$

(d) Of course, even without multiplying numerator and denominator of Q by $L - \sqrt{N}R$,

$$Q = N\left(\frac{L + (N/\sqrt{N})R}{L + (N/\sqrt{N})R}\right) = \sqrt{N}.$$

Moreover, $(sL + tR)/(L + sR) = s$ implies $sL + tR = sL + s^2R$ or $s = \sqrt{t}$; thus in one sense our accidents are unique!

7. ANOTHER MODIFICATION

Instead of considering the ratios L_k/R_k, suppose we now look at L_{k+1}/L_k. E.g., when $N = 2$ we get $L_6/L_5 = 239/99 \approx 2.41414$. Thus, in general, one might guess $L_{k+1}/L_k \to 1 + \sqrt{N}$ as $k \to \infty$. Not only is this the case in general, but in numerical analysis consideration of "ratios of corresponding components" yields the so-called Power Method for computing the numerically largest eigenvalue of a matrix.

To see the essential notions, let T be a 2×2 matrix with eigenvalues

$$|\lambda_1| > |\lambda_2| \geq 0$$

and linearly independent eigenvectors $\mathbf{x}_1$, $\mathbf{x}_2$. If $\mathbf{V}^{(0)}$ is an arbitrary vector, then suppose

$$\mathbf{V}^{(0)} = c_1\mathbf{x}_1 + c_2\mathbf{x}_2, \text{ where } c_1 \neq 0.$$

As before, define $\mathbf{V}^{(m)} = A\mathbf{V}^{(m-1)}$, $m = 1, 2, \ldots$. This yields

$$\mathbf{V}^{(m)} = c_1\lambda_1^m\mathbf{x}_1 + c_2\lambda_2^m\mathbf{x}_2 = \lambda_1^m\left(c_1\mathbf{x}_1 + c_2\left(\frac{\lambda_2}{\lambda_1}\right)^m\mathbf{x}_2\right)$$

(since $A\mathbf{x}_i = \lambda\mathbf{x}_i$) with the second term $\to 0$ as $m \to \infty$. Thus if $\mathbf{V}^{(m)} \approx \lambda_1^m c_1\begin{pmatrix} a \\ b \end{pmatrix}$, then the ratio of, say, first components

$$\frac{\lambda_1^{m+1}c_1a}{\lambda_1^m c_1 a}$$

approximates λ_1; moreover, $\begin{pmatrix} a \\ b \end{pmatrix}$ is a corresponding eigenvector.

In actual practice this version of the Power Method is usually improved by an appropriate scaling (such as normalization) to avoid overflow. Modifications for the case of a symmetric matrix and deflation techniques (for approximating nondominant eigenvalues) are discussed in [1].

8. CONCLUSION

It is somewhat amusing that for many years one of the authors asked students to investigate Algorithm II without being aware that its probable origins go back some nineteen centuries. An interesting discussion of its relationship to Pell's Equation as well as to the geometry of the ancient Greeks may be found in [3].

We leave the reader with at least two possible excursions. Suppose N is a positive (nonsquare) integer with continued fraction convergents p_k/q_k [2].

(a) If $L_k/R_k = p_k/q_k$ for $k = 0$ and 1, what can you say about N?

(b) If the equation in (a) holds for $k = 0$, 1, and 2, what can be said about N? (E.g., it holds when $N = 7$.)

REFERENCES

1. R. L. Burden, J. D. Faires, & A. C. Reynolds. *Numerical Analysis*. Prindle, Weber and Schmidt, 1978.
2. C. D. Olds. *Continued Fractions*. Random House, 1963.
3. D. Shanks. *Solved and Unsolved Problems in Number Theory*. Spartan Books, 1962.

*****