AN ALGORITHM FOR DETERMINING $R(N)$ FROM THE SUBSCRIPTS OF THE ZECKENDORF REPRESENTATION OF N

David A. Englund
9808 Wilcox Drive, Belvidere, IL 61008
(Submitted April 1999-Final Revision July 1999)

1. INTRODUCTION

Let $R(N)$ be the number of representations of the positive integer N as the sum of distinct Fibonacci numbers. N has a unique Zeckendorf representation [4], [3], in which no two consecutive Fibonacci numbers appear in the sum. Several methods have been developed for determining $R(N)$, many of which involve recursive formulas based on the number of representations of smaller integers [1]. In this paper we present an algorithm for determining $R(N)$ solely from the subscripts of the Zeckendorf representation of N. Carlitz [2, p. 210] has given a similar algorithm that can be used in the special case in which the subscripts in the Zeckendorf representation have the same parity.

2. STATEMENT OF THE ALGORITHM

Algorithm for $R(N)$: Write the Zeckendorf representation of N with the subscripts in descending order as follows:

$$N = \sum_{i=1}^{t} F(S_{t+1-i}) = F(S_t) + F(S_{t-1}) + F(S_{t-2}) + \cdots + F(S_j) + F(S_{j-1}) + \cdots + F(S_1),$$

where $S_j \geq S_{j-1} + 2$ and $S_1 \geq 2$, and $F(k) = F_k$. Define:

$T_0 = 1$;

$T_1 = [S_1/2]$ (where [ ] is the greatest integer function). Let

$T_j = [(S_j - S_{j-1} + 2)/2]T_{j-1}$ if S_j and S_{j-1} are of opposite parity;

$T_j = [(S_j - S_{j-1} + 2)/2]T_{j-1} - T_{j-2}$ if S_j and S_{j-1} are of the same parity.

Then $R(N) = T_t$.

Example 1: Find $R(63)$. The Zeckendorf representation of $63 = F_{10} + F_6$. Thus:

$T_0 = 1$ (by definition);

$T_1 = [6/2] = 3$;

$T_2 = [(10-6+2)/2]T_1 - T_0 = (3)(3) - 1 = 8 = R(63)$.

Example 2: Find $R(824)$. The Zeckendorf representation of $824 = F_{15} + F_{12} + F_{10} + F_7 + F_3$. Thus:

$T_0 = 1$ (by definition);

$T_1 = [3/2] = 1$;

$T_2 = [(7-3+2)/2]T_1 - T_0 = (3)(1) - 1 = 2$;

$T_3 = [(10-7+2)/2]T_2 = (2)(2) = 4$;

$T_4 = [(12-10+2)/2]T_3 - T_2 = (2)(4) - 2 = 6$;

$T_5 = [(15-12+2)/2]T_4 = (2)(6) = 12 = R(824)$.

Remark: In the special case in which all S_i are even, the validity of the present algorithm follows easily from the algorithm of Carlitz [2, p. 210]. (Alternatively, the validity of the algorithm of Carlitz follows from the validity of the present algorithm.) Suppose that all S_i are even. We write $S_t = 2k_1,\ S_{t-1} = 2k_2, \ldots, S_1 = 2k_t$, and $j_s = k_s - k_{s+1},\ s = 1, \ldots, t-1,\ j_t = k_t$. Still following Carlitz, we define $C_0 = 1,\ C_1 = j_1 + 1$, and $C_s = (j_s + 1)C_{s-1} - C_{s-2},\ s = 2, \ldots, t$. Slightly modifying the last step, we define $C_t' = j_t C_{t-1} - C_{t-2}$. Writing

$$C(x_1, \ldots, x_t) = \begin{vmatrix} x_1 & -1 & 0 & \cdots & 0 \\ -1 & x_2 & -1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & x_t \end{vmatrix}$$

for the continuant, we have [2, p. 212]

$$\begin{aligned} R(N) &= C_t - C_{t-1} = C_t' \\ &= C(j_1 + 1, j_2 + 1, \ldots, j_{t-1} + 1, j_t) \\ &= C(j_t, j_{t-1} + 1, j_{t-2} + 1, \ldots, j_1 + 1) = T_t. \end{aligned}$$

For example, if $N = F_{16} + F_8 + F_4 = 1011$, then $(j_1 + 1, j_2 + 1, j_3) = (5, 3, 2)$ and the (modified) Carlitz algorithm gives:

$C_0 = 1;$

$C_1 = 5;$

$C_2 = (3)(5) - 1 = 14;$

$C_3' = (2)(14) - 5 = 23 = R(N).$

Using the present algorithm, we obtain:

$T_0 = 1;$

$T_1 = 2;$

$T_2 = (3)(2) - 1 = 5;$

$T_3 = (5)(5) - 2 = 23 = R(N).$

3. PROOF OF THE ALGORITHM

Lemma: Following the steps of the algorithm set forth in Section 2, if $N = F_m - 1$ $(m \geq 3)$, then $T_0 = T_1 = \cdots = T_t = 1$.

Proof: This follows immediately from the formulas [3]

$$F_3 + F_5 + \cdots + F_{2n-1} = F_{2n} - 1 \ (n \geq 2) \ \text{and} \ F_2 + F_4 + \cdots + F_{2n} = F_{2n+1} - 1 \ (n \geq 1).$$

Theorem: Following the steps of the algorithm set forth in Section 2, $R(N) = T_t$.

Proof: We use induction on t, the number of terms in the Zeckendorf representation of N. (Note that, if $t > 1$, then the Zeckendorf representation of $N - F(S_t)$ is clearly $F(S_{t-1}) + F(S_{t-2}) + \cdots + F(S_j) + F(S_{j-1}) + \cdots + F(S_1)$.)

1. The cases $t = 1$ and $t = 2$ follow immediately from the formula [1, p. 53] $R(F_n) = [n/2]$ and from [1, Theorem 7, p. 58], respectively.

2. We suppose now that $t \geq 3$ and that the theorem is valid for $t-1$ and $t-2$. Let $S_t = m$ and $S_{t-1} = n$, so that $m - n \geq 2$ and $n \geq 4$. We write

$$N' = N - F(S_t) = F(S_{t-1}) + F(S_{t-2}) + \cdots + F(S_j) + F(S_{j-1}) + \cdots + F(S_1).$$

Then we have $F_n \leq N' \leq F_{n+1} - 1$.

a) If $F_n \leq N' \leq F_{n+1} - 2$, we use [1, Corollary 3.1, p. 53].

a-1) Suppose that $m-n$ is odd. Then

$$R(n) = [(m-n+1)/2]R(N') = [(m-n+2)/2]T_{t-1} = T_t,$$

using the induction hypothesis.

a-2) Suppose that $m-n$ is even. Using [1, Theorem 2, p. 48] and the induction hypothesis, we get

$$\begin{aligned} R(N) &= [(m-n+1)/2]R(N') + R(F_{n+1} - 2 - N') \\ &= [(m-n+2)/2]R(N') - (R(N') - R(F_{n+1} - 2 - N')) \\ &= [(m-n+2)/2]R(N') - R(N' - F_n) = [(m-n+2)/2]T_{t-1} - T_{t-2} = T_t. \end{aligned}$$

b) Suppose now that $N' = F_{n+1} - 1 = F_n + F_{n-2} + \cdots$. By [1, Theorem 7, p. 58], we have $R(N) = [(m-n+1)/2]$. On the other hand, using the Lemma, we have

$$T_t = ([(m-n+2)/2])(1) = [(m-n+1)/2]$$

if $m-n$ is odd, while

$$T_t = ([(m-n+2)/2])(1) - 1 = [(m-n+1)/2]$$

if $m-n$ is even. So we have $R(N) = T_t$ in this case also. This completes the proof.

ACKNOWLEDGMENT

The author wishes to thank the anonymous referee for useful suggestions and for a proof of the algorithm by induction, which is considerably shorter than the proof in graph form contained in the author's original submission.

REFERENCES

1. M. Bicknell-Johnson & D. C. Fielder. "The Number of Representations of N Using Distinct Fibonacci Numbers, Counted by Recursive Formulas." *The Fibonacci Quarterly* **37.1** (1999): 47-60.
2. L. Carlitz. "Fibonacci Representations." *The Fibonacci Quarterly* **6.4** (1968):193-220.
3. V. E. Hoggatt, Jr. *Fibonacci and Lucas Numbers*. Boston: Houghton Mifflin, 1969; rpt. Santa Clara, CA: The Fibonacci Association, 1979.
4. E. Zeckendorf. "Representation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas." *Bull. Soc. Royale Sci. Liège* **41** (1972):179-82,

AMS Classification Numbers: 11B39, 11B37, 11Y55

❖❖❖