UNARY FIBONACCI NUMBERS ARE CONTEXT-SENSITIVE

Vamsi K. Mootha (student)
Stanford University, Stanford, CA 94305
(Submitted April 1991)

Moll and Venkatesan showed in [2] that the set of Fibonacci numbers is not context-free. Recall that a language is CF (context-free) if and only if there exists a context-free grammar generating it. It is only natural to ask where exactly in Chomsky's Hierarchy the Fibonacci numbers lie. By the Hierarchy Theorem (Theorem 9.9 of [1]), we have the following proper containments:

Regular sets $\subset$ CFL's $\subset$ CSL's $\subset$ RE's

RE's (recursively enumerable languages) are defined to be those sets generated by unrestricted grammars. Unrestricted grammars are simply grammars in which all the productions are of the form $\alpha \to \beta$, where α and β are arbitrary strings of grammar symbols, with $\alpha \neq \varepsilon$. By definition, CSL's (context-sensitive languages) are generated by CSG's (context-sensitive grammars). CSG's are very much like unrestricted grammars, with the added condition that for all productions $\alpha \to \beta$, $|\alpha| \le |\beta|$.

In this paper we offer a CSG G generating the language of unary Fibonacci numbers, $L = \{0^i \mid i = F_n\}$, hence demonstrating the title claim. But before doing this, it will prove useful to construct an unrestricted grammar G' for L.

THE UNRESTRICTED GRAMMAR G'

Formally define $G' = (V', T, P', S)$, where $V' = \{S, A, B, C, D, E, F, G, H, J, K, L, M, N, P\}$, $T = \{0\}$, and P' is given by the list of productions:

1)	$S \to 0$	14)	$KC \to LC0$
2)	$S \to AE0B0CD$	15)	$0L \to L0$
3)	$AE \to AH$	16)	$BL \to BJ$
4)	$H0 \to F0$	17)	$BJC \to BM$
5)	$F00 \to 0F0$	18)	$M0 \to 0M$
6)	$F0B \to BF0$	19)	$MD \to NCD$
7)	$F0C \to GC0$	20)	$0N \to N0$
8)	$0G \to G0$	21)	$BN \to NB$
9)	$BG \to GB$	22)	$AN \to AE$
10)	$AG \to AH$	23)	$AE \to P$
11)	$AHB \to ABJ$	24)	$P0 \to 0P$
12)	$BJ0 \to 0BK$	25)	$PB \to P$
13)	$K0 \to 0K$	26)	$PCD \to \varepsilon$

Observe that there are two starting productions. Production 1 generates the nonrecursive base cases; production 2 generates all other Fibonacci numbers F_n, with $n > 2$. In general selection of production 3 eventually leads to a string of the form

$(*) \qquad AE0\ldots0B0\ldots0CD.$

The 0's between A and B represent unary F_{n-2}, while those between B and C represent F_{n-1}. Repeated selection of production 3 "increments" $(*)$, while choosing production 23 outputs F_n by eliminating the markers.

In summary, productions 1 and 2 enable us to generate either the base or recursive case. Productions 3 through 11 move F_{n-2} into the space between C and D; productions 12 through 22 perform the updating and restoration of the string to the form of $(*)$. Finally, productions 23 to 26 output the answer. It is easily verified that G' generates exactly L. □

Because G' is an unrestricted grammar that generates L, L is recursively-enumerable. Note that G' is not a CSG because the left-hand sides of productions 23, 25, and 26 are longer than their right-hand sides.

THE CONTEXT-SENSITIVE GRAMMAR *G*

We use the method of Example 9.5 of [1] to create a context-sensitive grammar G which mimics G'. Instead of the "single" variables of G', we use "composite" variables that combine 0 with each of its possible contexts. For example, the single nonterminal $[AE0]$ replaces the two variable string AE in a particular context.

Formally define $G=(V,T,P,[S])$, where $V=$ {$[S]$, $[AE0]$, $[B0CD]$, $[AH0]$, $[AF0]$, $[ABF0]$, $[0CD]$, $[AB0]$, $[F0CD]$, $[F0]$, $[A0]$, $[BF0]$, $[B0]$, $[GC0D]$, $[C0D]$, $[GC0]$, $[0D]$, $[ABG0]$, $[G0]$, $[BG0]$, $[GB0]$, $[AG0]$, $[C0]$, $[AGB0]$, $[AHB0]$, $[ABJ0]$, $[BKC0D]$, $[BK0]$, $[BJ0]$, $[BKC0]$, $[KC0D]$, $[K0]$, $[KC0]$, $[BLC0]$, $[LC0]$, $[BL0]$, $[L0]$, $[BJC0]$, $[BM0]$, $[M0D]$, $[0MD]$, $[M0]$, $[0NCD]$, $[N0CD]$, $[BN0]$, $[0CD]$, $[NB0]$, $[AN0]$, $[N0]$, $[P0]$, $[PB0]$, $[P0CD]$, $[0PCD]$}, and P is given by the following list of productions, which are grouped according to the production of G' they mimic:

1) $[S] \to 0$

2) $[S] \to [AE0][B0CD]$

3) $[AE0] \to [AH0]$

4) $[AH0] \to [AF0]$

5) $[ABF0][0CD] \to [AB0][F0CD]$
$[ABF0]0 \to [AB0][F0]$
$[F0][0CD] \to 0[F0CD]$
$[AF0]0 \to [A0][F0]$
$[BF0]0 \to [B0][F0]$
$[F0]0 \to 0[F0]$

6) $[AF0][B0CD] \to [ABF0][0CD]$
$[AF0][B0] \to [ABF0]0$
$[F0][B0] \to [BF0]0$

7) $[F0CD] \to [GC0D]$
$[F0][C0D] \to [GC0][0D]$

8) $[AB0][GC0D] \to [ABG0][C0D]$
$0[GC0D] \to [G0][C0D]$
$[AB0][G0] \to [ABG0]0$
$0[G0] \to [G0]0$
$[B0][G0] \to [BG0]0$
$[A0][GB0] \to [AG0][B0]$
$0[GC0] \to [G0][C0]$

14) $[BKC0D] \to [BLC0][0D]$
$[KC0D] \to [LC0][0D]$
$[BKC0] \to [BLC0]0$
$[KC0] \to [LC0]0$

15) $[B0][LC0] \to [BL0][C0]$
$0[LC0] \to [L0][C0]$
$[B0][L0] \to [BL0]0$

16) $[BLC0] \to [BJC0]$
$[BL0] \to [BJ0]$

17) $[BJC0] \to [BM0]$

18) $[BM0][0D] \to [B0][M0D]$
$[M0D] \to [0MD]$
$[BM0]0 \to [B0][M0]$
$[M0][0D] \to 0[M0D]$
$[M0]0 \to 0[M0]$

19) $[0MD] \to [0NCD]$

20) $[0NCD] \to [N0CD]$
$[B0][N0CD] \to [BN0][0CD]$
$[A0][NB0] \to [AN0][B0]$
$0[N0CD] \to [N0][0CD]$
$[B0][N0] \to [BN0]0$
$0[NB0] \to [N0][B0$
$[A0][N0] \to [AN0]0$
$0[N0] \to [N0]0$

9) $[ABG0] \to [AGB0]$
$[BG0] \to [GB0]$

10) $[AGB0] \to [AHB0]$
$[AG0] \to [AH0]$

11) $[AHB0] \to [ABJ0]$

12) $[ABJ0][C0D] \to [A0][BKC0D]$
$[ABJ0]0 \to [A0][BK0]$
$[A0][BJ0][C0] \to [A0]0[BKC0]$
$[BJ0]0 \to 0[BK0]$
$[BJ0][C0] \to 0[BKC0]$

13) $[BK0][C0D] \to [B0][KC0D]$
$[BK0]0 \to [B0][K0]$
$[K0][C0] \to 0[KC0]$
$[BK0][C0] \to [B0][KC0]$
$[K0]0 \to 0[K0]$

21) $[BN0] \to [NB0]$

22) $[AN0] \to [AE0]$

23) $[AE0] \to [P0]$

24) $[P0]0 \to 0[P0]$
$[P0][B0] \to 0[PB0]$
$[P0][0CD] \to 0[P0CD]$
$[P0CD] \to [0PCD]$

25) $[PB0] \to [P0]$

26) $[0PCD] \to 0$

It is straightforward to see that $S \overset{*}{\Rightarrow} \alpha'$ (i.e., a string α' is derived from S) through G' if and only if $[S] \overset{*}{\Rightarrow} \alpha$ with G, where α is formed from α' by grouping with a 0 all markers (i.e., elements of $V' - \{S\}$) appearing between it and the 0 to its left, and also by grouping the first 0 with any markers to its left and with the last 0 any markers to its right; e.g., if α' is $A00B0KC000D$, then α is $[A0]0[B0][KC0]0[0D]$. Observe that the right side of every production of G is at least as long as the left side. Clearly, G is a context-sensitive grammar. □

Thus, we have

Theorem: L is a context-sensitive language.

Proof: Immediate from construction of G. □

REFERENCES

1. J. Hopcroft & J. Ullman. *Introduction to Automata Theory, Languages, and Computation.* New York: Addison-Wesley, 1979.
2. R. Moll & S. Venkatesan. "Fibonacci Numbers are Not Context-Free." *Fibonacci Quarterly* **29.1** (1991):59-61.

AMS Classification Numbers: 68Q50, 68Q45

❖❖❖