Suppose we add laws to allow various operators to distribute over string join (semi-colon). For example, if \( i \) and \( j \) are items and \( s \) and \( t \) are strings, then the laws
\[
\text{nil} + \text{nil} = \text{nil}
\]
\[
(i; s) + (j; t) = i+j; s+t
\]
say that strings are added item by item (a sum of strings is a string of sums). For example,
\[
(2; 4; 7) + (3; 9; 1) = 5; 13; 8
\]
What string \( f \) is defined by
\[
f = 0; 1; f+f_{1:..\infty}
\]

§ Using \( \ldots \) for unknown items, here are \( f \) and \( f_{1:..\infty} \) and below that their sum.
\[
f = 0; 1; \ldots
\]
\[
f_{1:..\infty} = 1; \ldots
\]
\[
f+f_{1:..\infty} = 1; 2; \ldots
\]
So now we know item 2. So again
\[
f = 0; 1; 1; \ldots
\]
\[
f_{1:..\infty} = 1; 1; \ldots
\]
\[
f+f_{1:..\infty} = 1; 2; 1; \ldots
\]
So now we know item 3. So again
\[
f = 0; 1; 1; 2; \ldots
\]
\[
f_{1:..\infty} = 1; 1; 2; \ldots
\]
\[
f+f_{1:..\infty} = 1; 2; 3; \ldots
\]
So now we know item 4. And so on. It's the Fibonacci sequence
\[
f = 0; 1; 1; 2; 3; 5; 8; 13; 21; 34; \ldots
\]
usually defined as
\[
f_0 = 0
\]
\[
f_1 = 1
\]
\[
f_{n+2} = f_{n} + f_{n+1}
\]