EXCERPTED FROM

STEPHEN WOLFRAM
A NEW KIND OF SCIENCE

SECTION 4.2

Elementary Arithmetic

0 through 9. But as the picture at the bottom of the facing page shows, one can equally well use other bases. And in practical computers, for example, base 2 is almost always what is used.

So what this means is that in a computer numbers are represented by sequences of 0's and 1's, much like sequences of white and black cells in systems like cellular automata. And operations on numbers then correspond to ways of updating sequences of 0's and 1's.

In traditional mathematics, the details of how operations performed on numbers affect sequences of digits are usually considered quite irrelevant. But what we will find in this chapter is that precisely by looking at such details, we will be able to see more clearly how complexity develops in systems based on numbers.

In many cases, the behavior we find looks remarkably similar to what we saw in the previous chapter. Indeed, in the end, despite some confusing suggestions from traditional mathematics, we will discover that the general behavior of systems based on numbers is very similar to the general behavior of simple programs that we have already discussed.

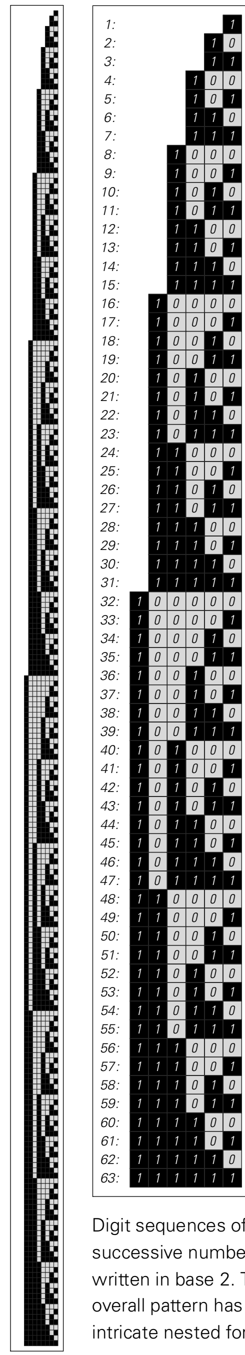
Elementary Arithmetic

The operations of elementary arithmetic are so simple that it seems impossible that they could ever lead to behavior of any great complexity. But what we will find in this section is that in fact they can.

To begin, consider what is perhaps the simplest conceivable arithmetic process: start with the number 1 and then just progressively add 1 at each of a sequence of steps.

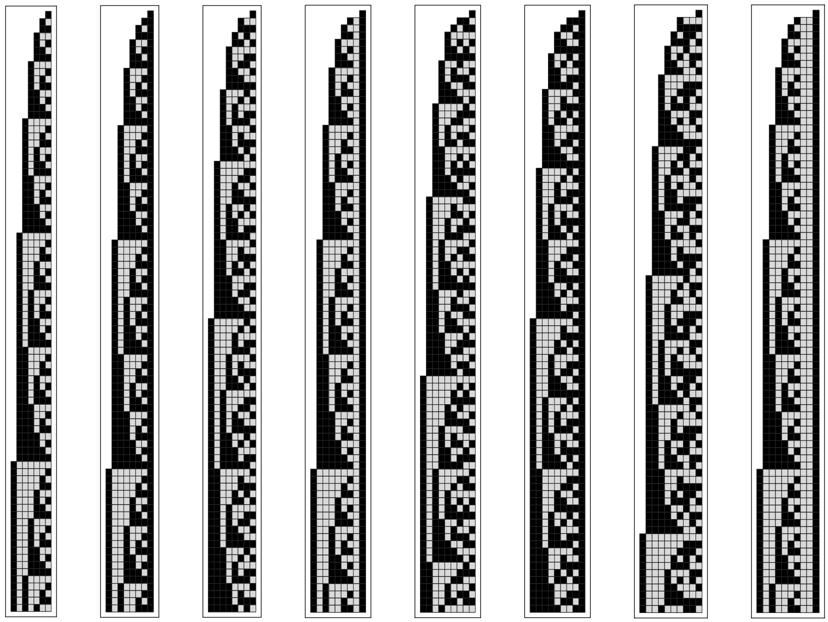
The result of this process is to generate the successive numbers 1, 2, 3, 4, 5, 6, 7, 8, ... The sizes of these numbers obviously form a very simple progression.

But if one looks not at these overall sizes, but rather at digit sequences, then what one sees is considerably more complicated. And in fact, as the picture on the right demonstrates, these successive digit sequences form a pattern that shows an intricate nested structure.



Digit sequences of successive numbers written in base 2. The overall pattern has an intricate nested form.

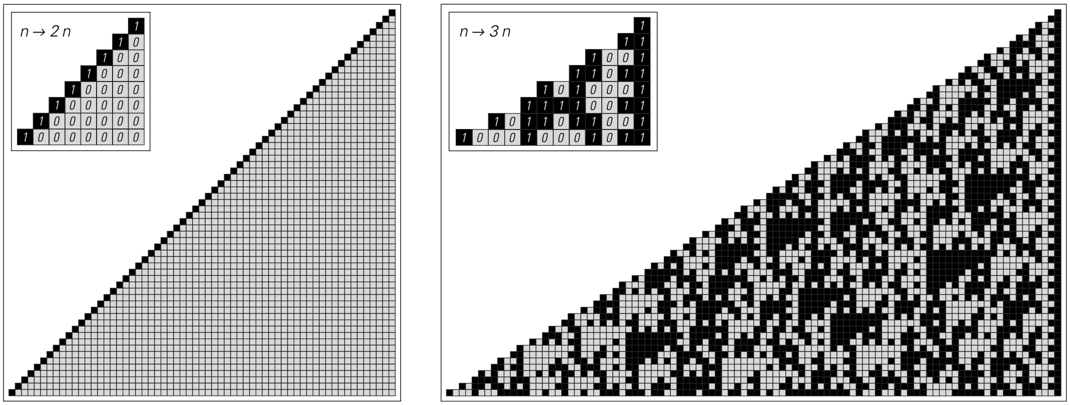
The pictures below show what happens if one adds a number other than 1 at each step. Near the right-hand edge, each pattern is somewhat different. But at an overall level, all the patterns have exactly the same basic nested structure.



Digit sequences in base 2 of numbers obtained by starting with 1 and then successively adding a constant at each step. All these patterns ultimately have the same overall nested form.

If instead of addition one uses multiplication, however, then the results one gets can be very different. The first picture at the top of the facing page shows what happens if one starts with 1 and then successively multiplies by 2 at each step.

It turns out that if one represents numbers as digit sequences in base 2, then the operation of multiplying by 2 has a very simple effect: it just shifts the digit sequence one place to the left, adding a 0 digit on the right. And as a result, the overall pattern obtained by successive multiplication by 2 has a very simple form.



Patterns produced by starting with the number 1, and then successively multiplying by a factor of 2, and a factor of 3. In each case, the digit sequence of the number obtained at each step is shown in base 2. Multiplication by 2 turns out to correspond just to shifting all digits in base 2 one position to the left, so that the overall pattern produced in this case is very simple. But multiplication by 3 yields a much more complicated pattern, as the picture on the right shows. Note that in these pictures the complete numbers obtained at each step correspond respectively to the successive integer powers of 2 and of 3.

But if the multiplication factor at each step is 3, rather than 2, then the pattern obtained is quite different, as the second picture above shows. Indeed, even though the only operation used was just simple multiplication, the final pattern obtained in this case is highly complex.

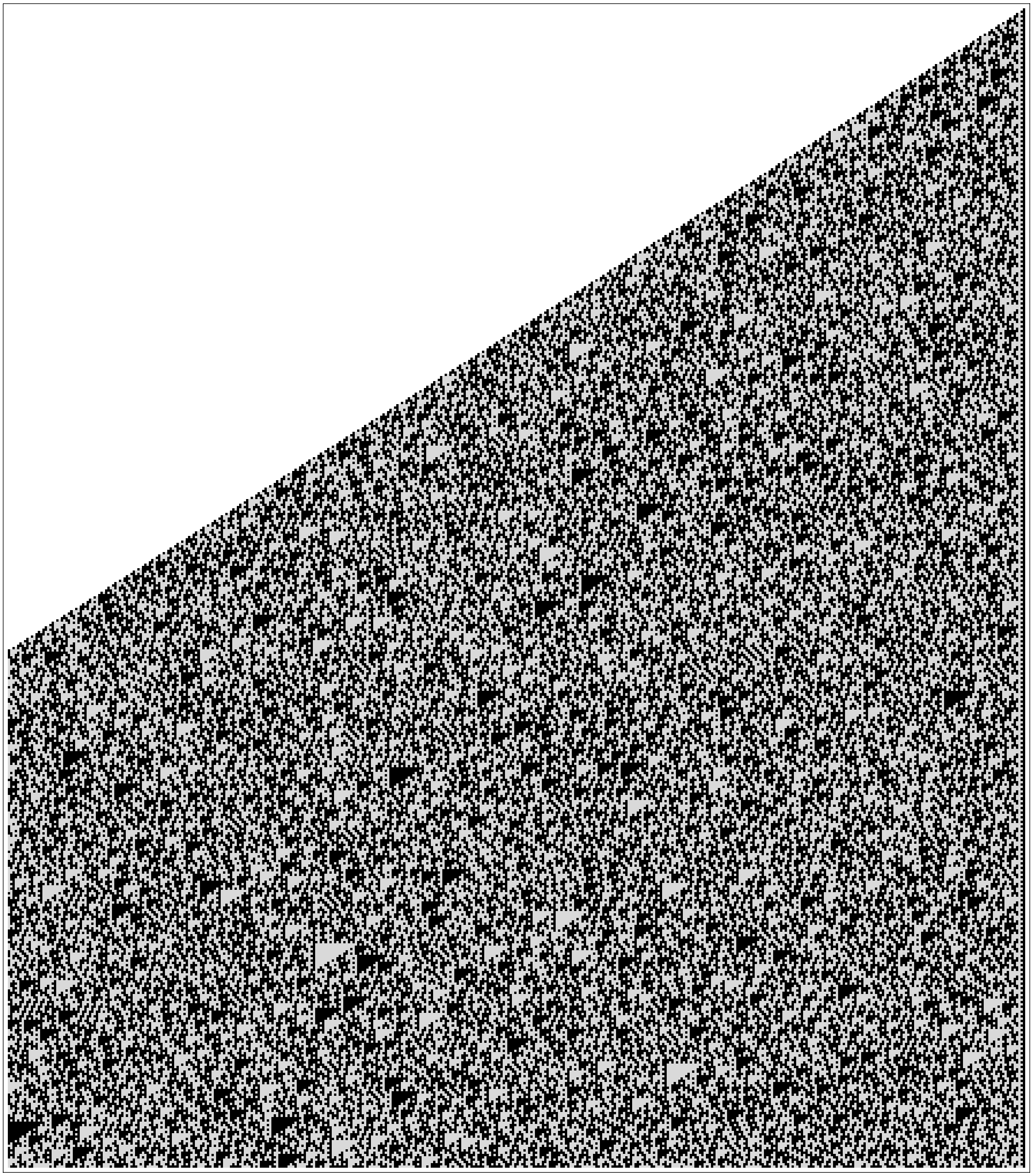
The picture on the next page shows more steps in the evolution of the system. At a small scale, there are some obvious triangular and other structures, but beyond these the pattern looks essentially random.

So just as in simple programs like cellular automata, it seems that simple systems based on numbers can also yield behavior that is highly complex and apparently random.

But we might imagine that the complexity we see in pictures like the one on the next page must somehow be a consequence of the fact that we are looking at numbers in terms of their digit sequences—and would not occur if we just looked at numbers in terms of their overall size.

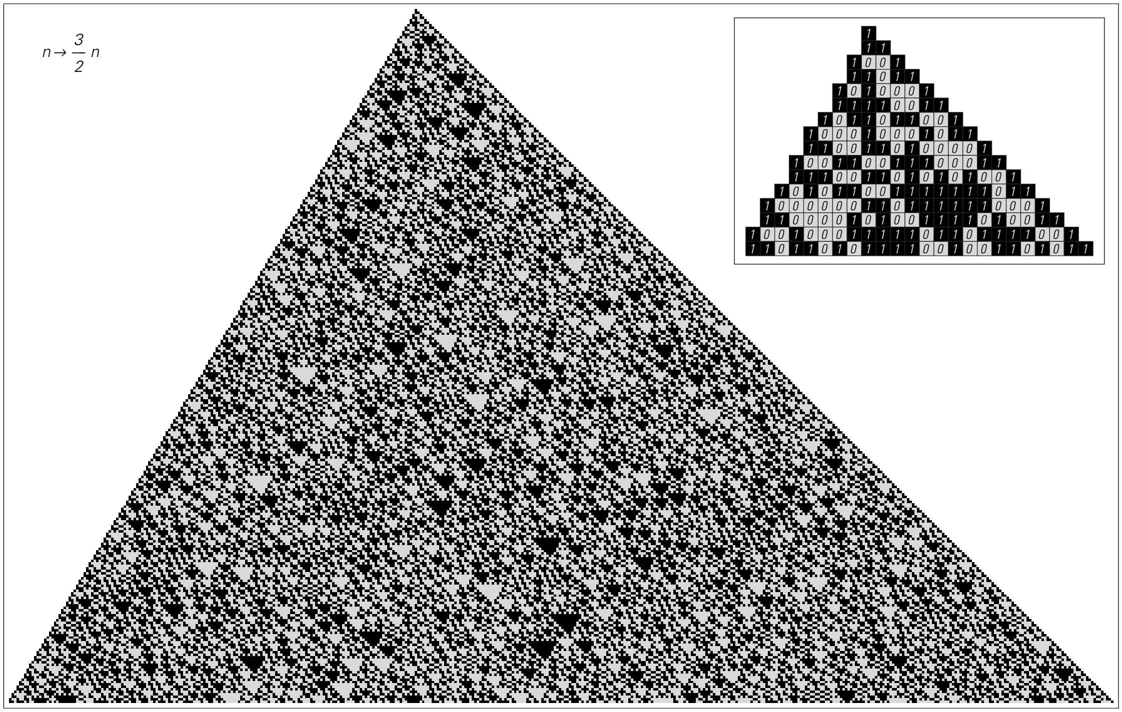
A few examples, however, will show that this is not the case.

To begin the first example, consider what happens if one multiplies by $3/2$, or 1.5, at each step. Starting with 1, the successive numbers that one obtains in this way are 1, $3/2 = 1.5$, $9/4 = 2.25$, $27/8 = 3.375$, $81/16 = 5.0625$, $243/32 = 7.59375$, $729/64 = 11.390625$, …



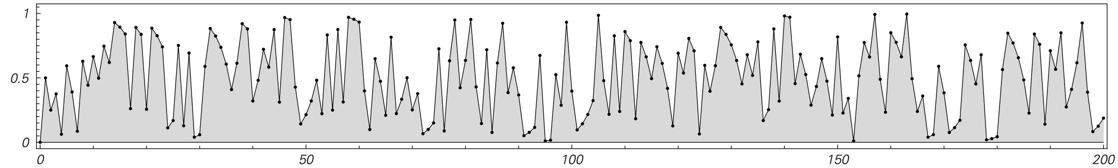
The first 500 powers of 3, shown in base 2. Some small-scale structure is visible, but on a larger scale the pattern seems for all practical purposes random. Note that the pattern shown here has been truncated at the edge of the page on the left, although in fact the whole pattern continues to expand to the left forever with an average slope of *Log[2, 3]* ≃ *1.58* .

The picture below shows the digit sequences for these numbers given in base 2. The digits that lie directly below and to the left of the original 1 at the top of the pattern correspond to the whole number part of each successive number (e.g. 3 in 3.375), while the digits that lie to the right correspond to the fractional part (e.g. 0.375 in 3.375).



Successive powers of 3/2, shown in base 2. Multiplication by 3/2 can be thought of as multiplication by 3 combined with division by 2. But division by 2 just does the opposite of multiplication by 2, so in base 2 it simply shifts all digits one position to the right. The overall pattern is thus a shifted version of the pattern shown on the facing page.

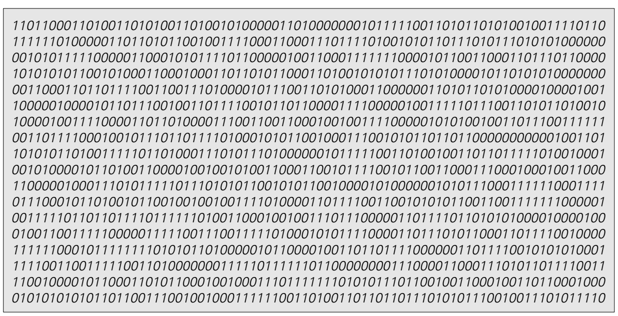
And instead of looking explicitly at the complete pattern of digits, one can consider just finding the size of the fractional part of each successive number. These sizes are plotted at the top of the next page. And the picture shows that they too exhibit the kind of complexity and apparent randomness that is evident at the level of digits.

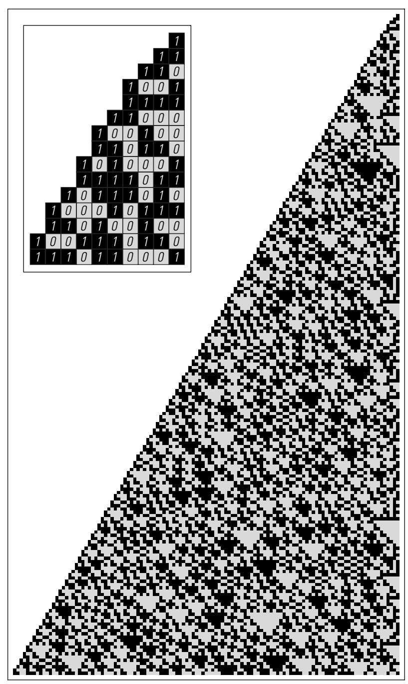


Sizes of the fractional parts of successive powers of 3/2. These sizes are completely independent of what base is used to represent the numbers. Only the dots are significant; the shading and lines between them are just included to make the plot easier to read.

The example just given involves numbers with fractional parts. But it turns out that similar phenomena can also be found in systems that involve only whole numbers.

As a first example, consider a slight variation on the operation of multiplying by $3/2$ used above: if the number at a particular step is even (divisible by 2), then simply multiply that number by $3/2$, getting a whole number as the result. But if the number is odd, then first add 1—so as to get an even number—and only then multiply by $3/2$.





Results of starting with the number 1, then applying the following rule: if the number at a particular step is even, multiply by 3/2; otherwise, add 1, then multiply by 3/2. This procedure yields a succession of whole numbers whose digit sequences in base 2 are shown at the right. The rightmost digits obtained at each step are shown above. The digit is 0 when the number is even and 1 when it is odd, and, as shown, the digits alternate in a seemingly random way. It turns out that the system described here is closely related to one that arose in studying the register machine shown on page 100. The system here can be represented by the rule $n \to$ *If[EvenQ[n], 3 n/2, 3 (n + 1)/2]*, while the one on page 100 follows the rule $n \to$ *If[EvenQ[n], 3 n/2, (3 n + 1)/2]*. After the first step these systems give the same sequence of numbers, except for an overall factor of 3.

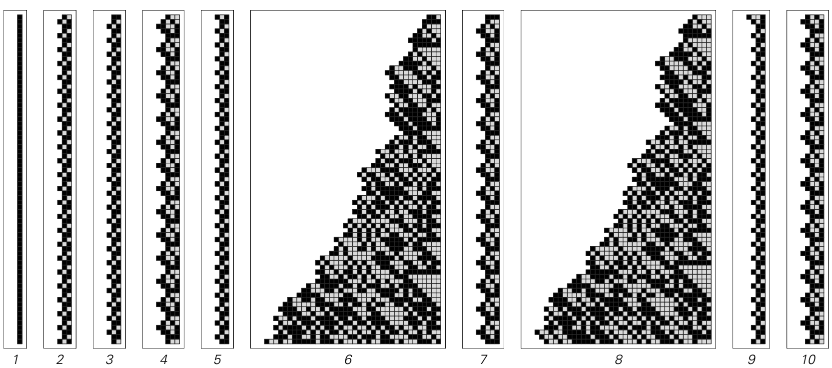
This procedure is always guaranteed to give a whole number. And starting with 1, the sequence of numbers one gets is 1, 3, 6, 9, 15, 24, 36, 54, 81, 123, 186, 279, 420, 630, 945, 1419, 2130, 3195, 4794, ...

Some of these numbers are even, while some are odd. But as the results at the bottom of the facing page illustrate, the sequence of which numbers are even and which are odd seems to be completely random.

Despite this randomness, however, the overall sizes of the numbers obtained still grow in a rather regular way. But by changing the procedure just slightly, one can get much less regular growth.

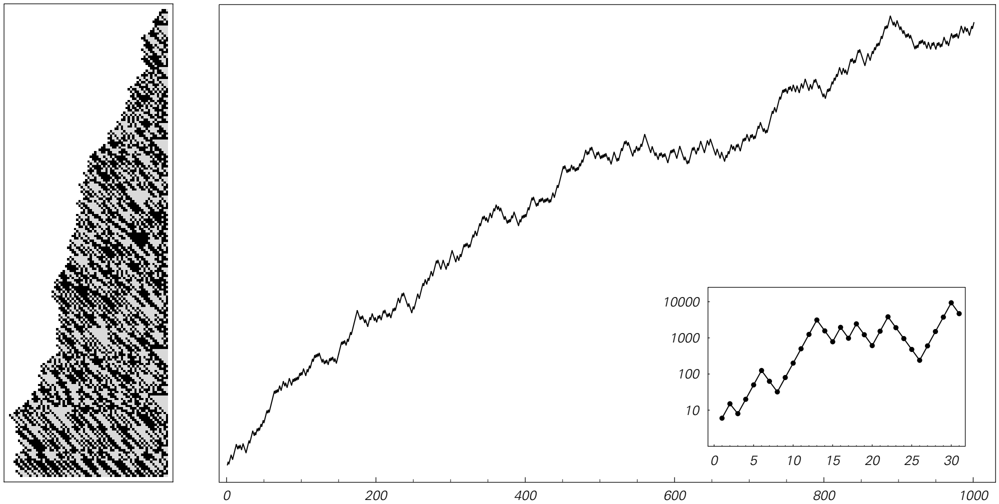
As an example, consider the following procedure: if the number obtained at a particular step is even, then multiply this number by $5/2$; otherwise, add 1 and then multiply the result by $1/2$.

If one starts with 1, then this procedure simply gives 1 at every step. And indeed with many starting numbers, the procedure yields purely repetitive behavior. But as the picture below shows, it can also give more complicated behavior.



Results of applying the rule *n → If[EvenQ[n], 5 n/2, (n + 1)/2]*, starting with different initial choices of *n*. In many cases, the behavior obtained is purely repetitive. But in some cases it is not.

Starting for example with the number 6, the sizes of the numbers obtained on successive steps show a generally increasing trend, but there are considerable fluctuations, and these fluctuations seem to be essentially random. Indeed, even after a million steps, when the



The results of following the same rule as on the previous page, starting from the value 6. Plotted on the right are the overall sizes of the numbers obtained for the first thousand steps. The plot is on a logarithmic scale, so the height of each point is essentially the length of the digit sequence for the number that it represents—or the width of the row on the left.

number obtained has 48,554 (base 10) digits, there is still no sign of repetition or of any other significant regularity.

So even if one just looks at overall sizes of whole numbers it is still possible to get great complexity in systems based on numbers.

But while complexity is visible at this level, it is usually necessary to go to a more detailed level in order to get any real idea of why it occurs. And indeed what we have found in this section is that if one looks at digit sequences, then one sees complex patterns that are remarkably similar to those produced by systems like cellular automata.

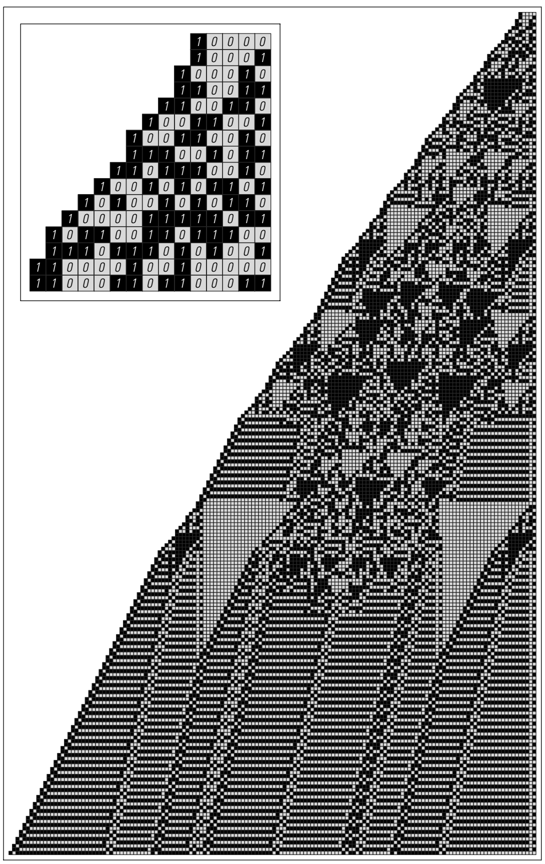
The underlying rules for systems like cellular automata are however usually rather different from those for systems based on numbers. The main point is that the rules for cellular automata are always local: the new color of any particular cell depends only on the previous color of that cell and its immediate neighbors. But in systems based on numbers there is usually no such locality.

One knows from hand calculation that even an operation such as addition can lead to "carry" digits which propagate arbitrarily far to the left. And in fact most simple arithmetic operations have the property that a digit which appears at a particular position in their result can depend on digits that were originally far away from it.

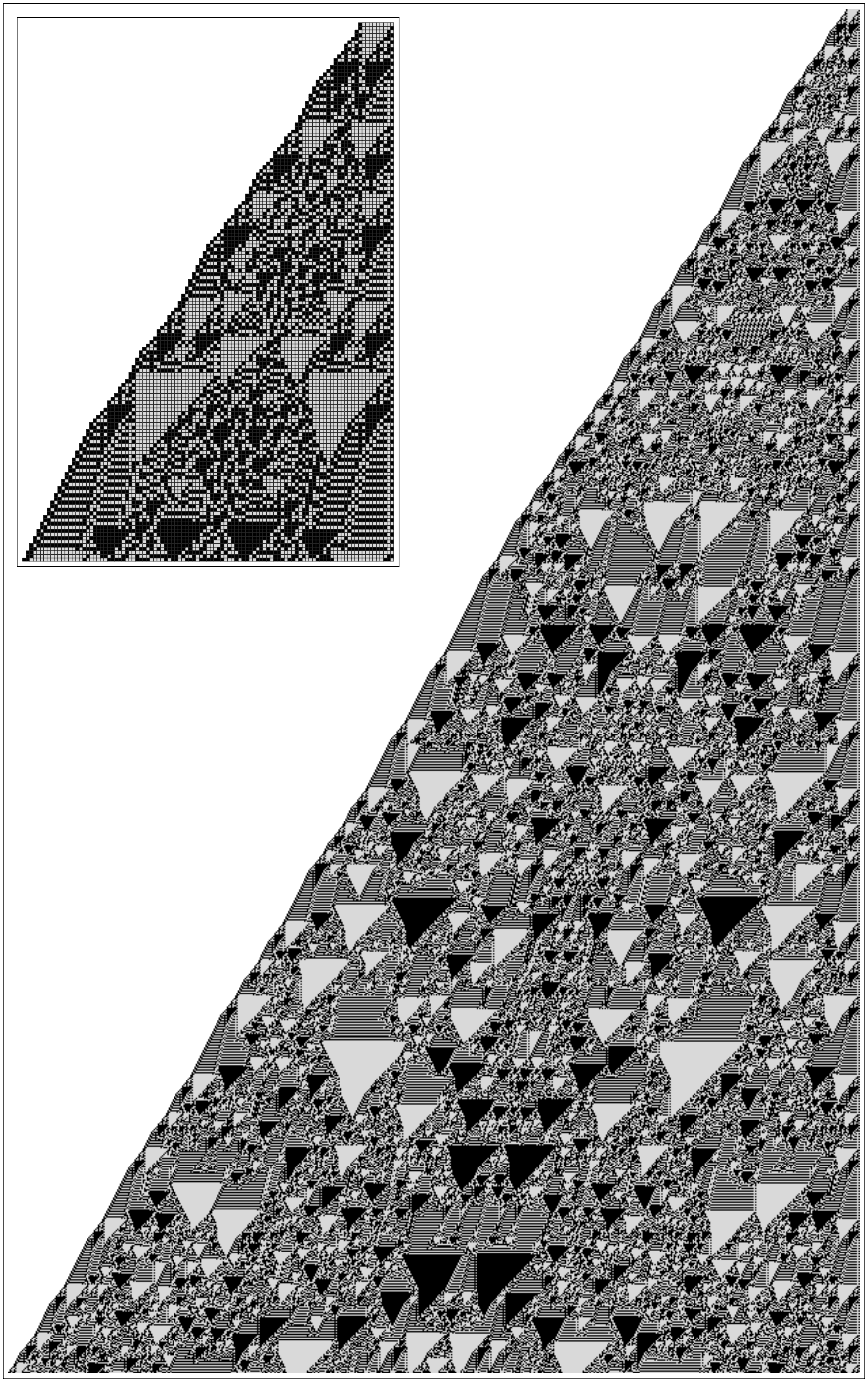
But despite fundamental differences like this in underlying rules, the overall behavior produced by systems based on numbers is still very similar to what one sees for example in cellular automata.

So just like for the various kinds of programs that we discussed in the previous chapter, the details of underlying rules again do not seem to have a crucial effect on the kinds of behavior that can occur.

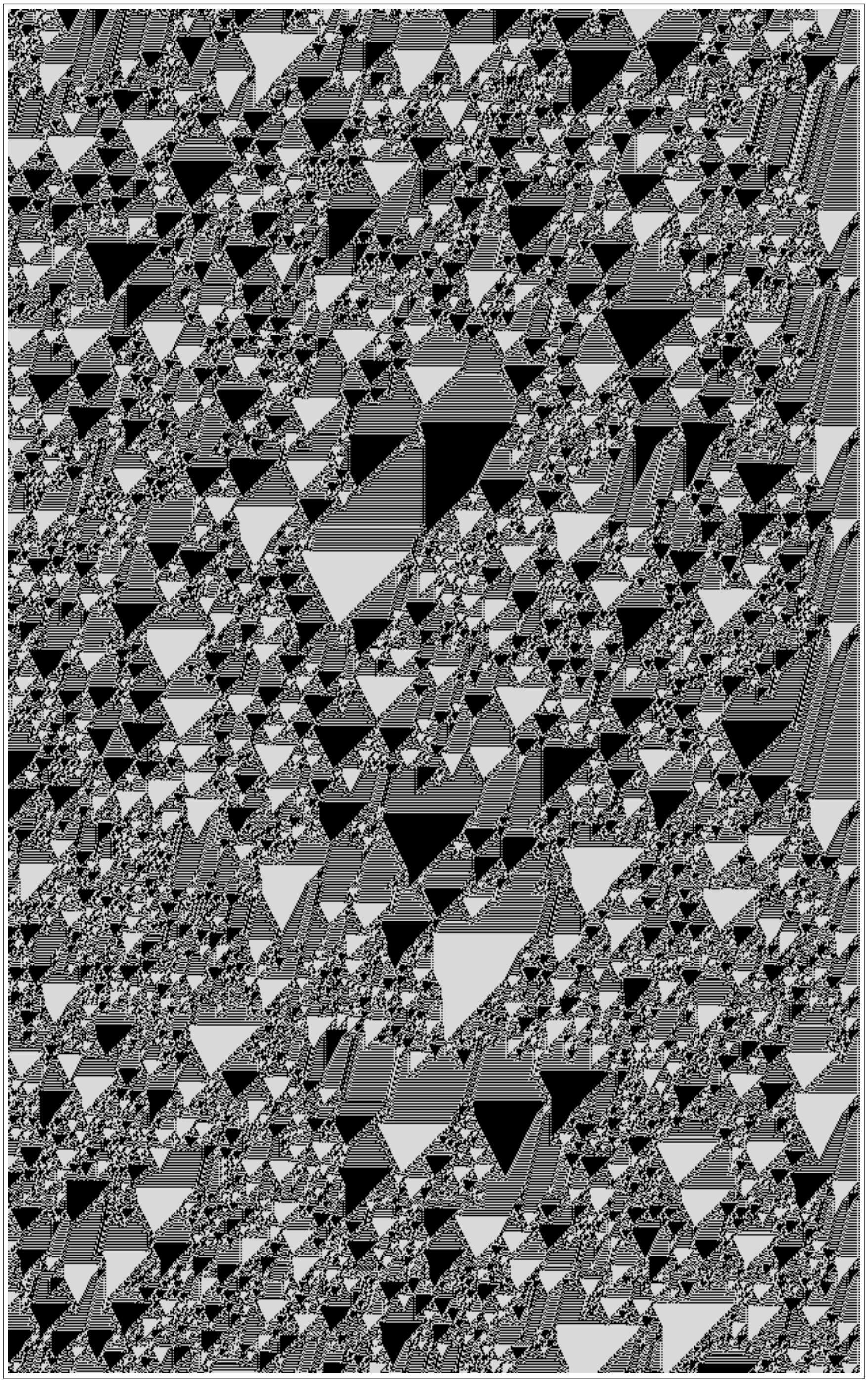
Indeed, despite the lack of locality in their underlying rules, the pictures below and on the pages that follow show that it is even possible to find systems based on numbers that exhibit something like the localized structures that we saw in cellular automata on page 32.



An example of a system defined by the following rule: at each step, take the number obtained at that step and write its base 2 digits in reverse order, then add the resulting number to the original one. For many possible starting numbers, the behavior obtained is very simple. This picture shows what happens when one starts with the number 16. After 180 steps, it turns out that all that survives are a few objects that one can view as localized structures.



A thousand steps in the evolution of a system with the same rule as on the previous page, but now starting with the number 512. Localized structures are visible, but the overall pattern never seems to take on any kind of simple repetitive form.



Continuation of the pattern on the facing page, starting at the millionth step. The picture shows the right-hand edge of the pattern; the complete pattern extends about 700 times the width of the page to the left.