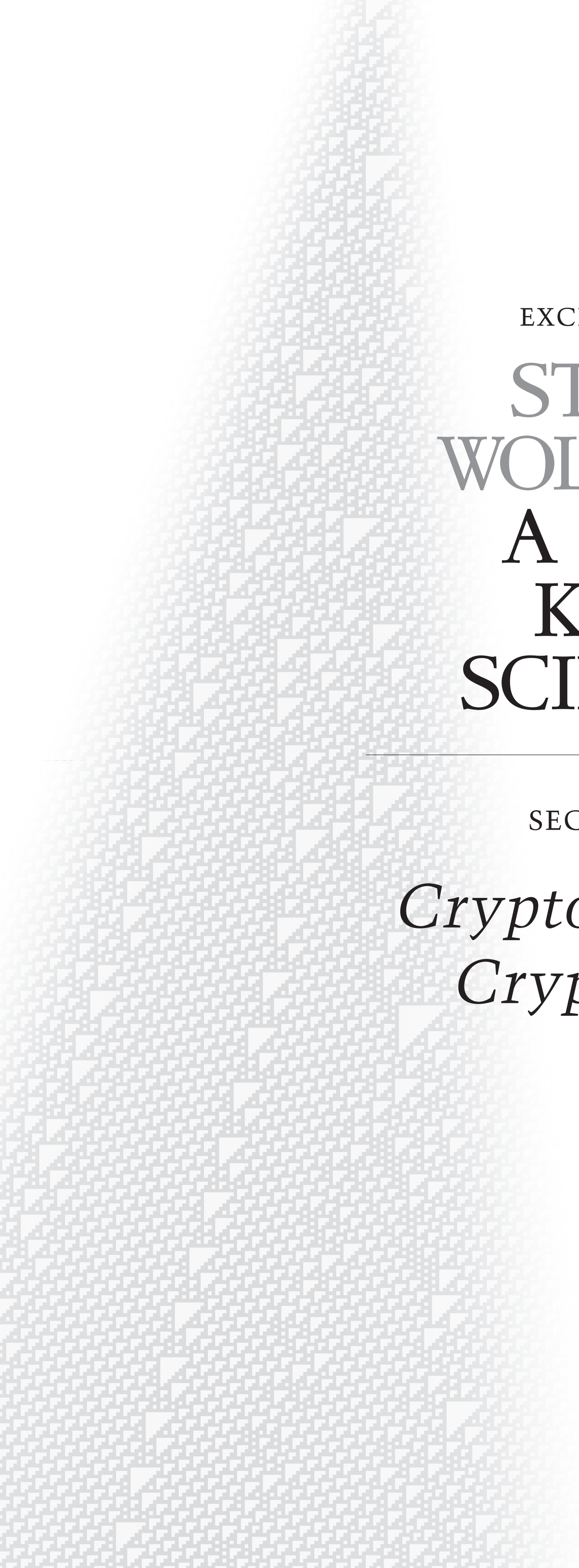


EXCERPTED FROM

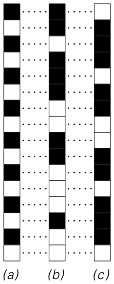
STEPHEN WOLFRAM
A NEW KIND OF SCIENCE

SECTION 10.10

Cryptography and Cryptanalysis

Cryptography and Cryptanalysis

The purpose of cryptography is to hide the contents of messages by encrypting them so as to make them unrecognizable except by someone who has been given a special decryption key. The purpose of cryptanalysis is then to defeat this by finding ways to decrypt messages without being given the key.



Example of a scheme for encryption. From the original message (a) an encrypted message (c) is generated by reversing the color of each square for which the corresponding square in the encrypting sequence (b) is black. This scheme is the basis for essentially all practical stream ciphers.

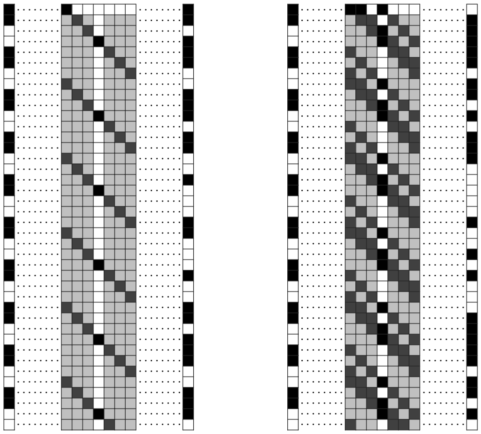
The picture on the left shows a standard method of encrypting messages represented by sequences of black and white squares. The basic idea is to have an encrypting sequence, shown as column (b) on the left, and from the original message (a) to get an encrypted version of the message (c) by reversing the color of every square for which the corresponding square in the encrypting sequence (b) is black.

So if one receives the encrypted message (c), how can one recover the original message (a)? If one knows the encrypting sequence (b) then it is straightforward. For all one need do is to repeat the process that was used for encryption, and reverse the color of every square in (c) for which the corresponding square in (b) is black.

But how can one arrange that only the intended recipient of the message knows the encrypting sequence (b)? In some situations it may be feasible to transmit the whole encrypting sequence in some secure way. But much more common is to be able to transmit only some short key in a secure way, and then to have to generate the encrypting sequence from this key.

So what kind of procedure might one use to get an encrypting sequence from a key? The picture at the top of the facing page shows an extremely simple approach that was widely used in practical cryptography until less than a century ago. The idea is just to form an encrypting sequence by repeatedly cycling through the elements in the key. And as the picture demonstrates, combining this with the original message leads to an encrypted message in which at least some of the structure in the original message is obscured.

But perhaps not surprisingly it is fairly easy to do cryptanalysis in such a case. For if one can find out what any sufficiently long segment in the encrypting sequence was, then this immediately gives the key,



A simple example of an encryption system in which the encrypting sequence is obtained by repetitively cycling through the elements of the key. Encryption with two different keys is shown. In each case the original message is on the left, the encrypted message is on the right, and the encrypting sequence corresponds to the highlighted column of cells. The system is essentially a Vigenère cipher of the kind widely used between the 1500s and the early 1900s.

and from the key the whole of the rest of the encrypting sequence can immediately be generated.

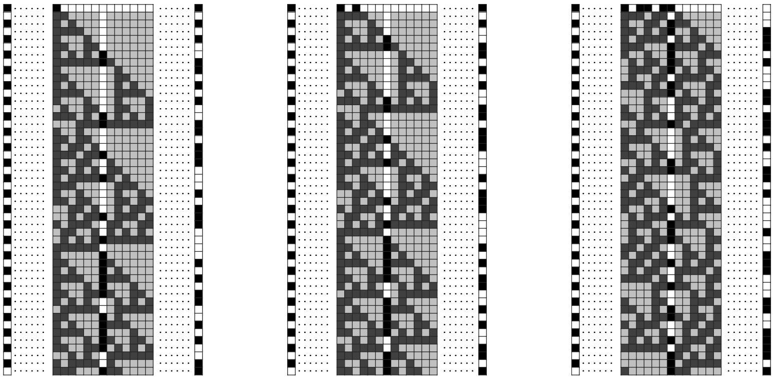
So what kind of analysis is needed to find a segment of the encrypting sequence? In an extreme but in practice common case one might happen to know what certain parts of the original message were—perhaps standardized greetings or some such—and by comparing the original and encrypted forms of these parts one can immediately deduce what the corresponding parts of the encrypting sequence must have been.

And even if all one knows is that the original message was in some definite language this is still typically good enough. For it means that there will be certain blocks—say corresponding to words like “the” in English—that occur much more often than others in the original message. And since such blocks must be encrypted in the same way whenever they occur at the same point in the repetition period of the encrypting sequence they will lead to occasional repeats in the encrypted message—with the spacing of such repeats always being some multiple of the repetition period. So this means that just by looking at the distribution of spacings between repeats one can expect to determine the repetition period of the encrypting sequence.

And once this is known, it is usually fairly straightforward to find the actual key. For one can pick out of the encrypted message all the squares that occur at a certain point in the repetition period of the encrypting sequence, and which are therefore encrypted using a particular element of the key. Then one can ask whether such squares are more often black or more often white, and one can compare this with the result obtained by looking at the frequencies of letters in the language of the original message. If these two results are the same, then it suggests that the corresponding element in the key is white, and if they are different then it suggests that it is black. And once one has found a candidate key it is easy to check whether the key is correct by trying to use it to recover some reasonably long part of the original message. For unless one has the correct key, the chance that what one recovers will be meaningful in the language of the original message is absolutely negligible.

So what happens if one uses a more complicated rule for generating an encrypting sequence from a key? Methods like the ones above still turn out to allow features of the encrypting sequence to be found. And so to make cryptography work it must be the case that even if one knows certain features or parts of the encrypting sequence it is still difficult to deduce the original key or otherwise to generate the rest of the sequence.

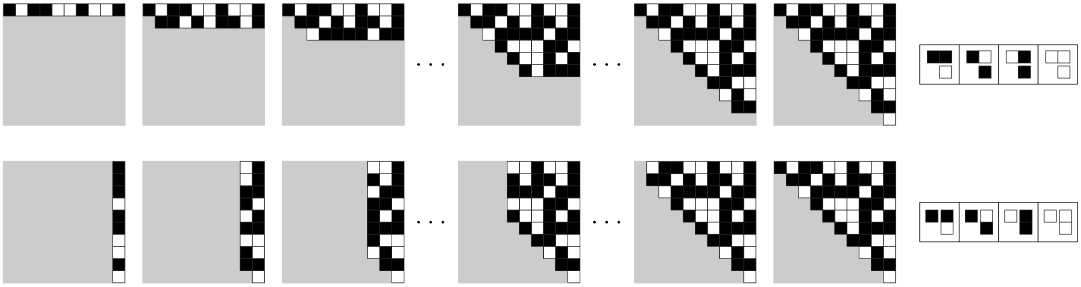
The picture below shows one way of generating encrypting sequences that was widely used in the early years of electronic cryptography, and is still sometimes used today. The basic idea is to look at the evolution of an additive cellular automaton in a register of limited width. The key then gives the initial condition for the cellular automaton, and the encrypting sequence is extracted, for example, by sampling a particular cell on successive steps.



Encryption using the rule 60 additive cellular automaton. This is essentially equivalent to a linear feedback shift register.

So given such an encrypting sequence, is there any easy way to do cryptanalysis and go backwards and work out the key?

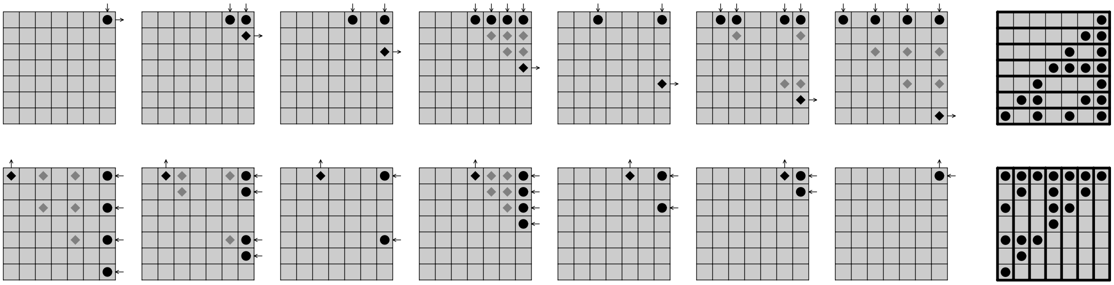
It turns out that there is. For as the picture below demonstrates, in an additive cellular automaton like the one considered here the underlying rule is such that it allows one not only to deduce the form of a particular row from the row above it, but also to deduce the form of a particular column from the column to its right. And what this means is that if one has some segment of the encrypting sequence, corresponding to part of a column, then one can immediately use this to deduce the forms of a sequence of other columns, and thus to find the form of a row in the cellular automaton—and hence the original key.

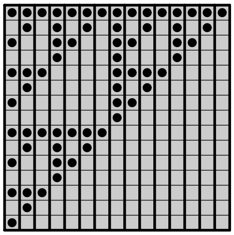


An example of the basis for cryptanalysis of an additive cellular automaton. The first set of pictures show the ordinary evolution of the rule 60 cellular automaton, in which each successive row is deduced from the one above. The second set of pictures show a kind of sideways evolution in which the rule is reinterpreted so as to allow a column of cells to be deduced from the column immediately to its right. Note that in both cases the colors of cells in the area on the lower right cannot be determined without knowing the colors of more initial cells than are shown.

But what happens if the encrypting sequence does not include every single cell in a particular column? One cannot then immediately use the method described above. But it turns out that the additive nature of the underlying rule still makes comparatively straightforward cryptanalysis possible.

The picture on the next page shows how this works. Because of additivity it turns out that one can deduce whether or not some cell a certain number of steps down a given column is black just by seeing whether there are an odd or even number of black cells in certain specific positions in the row at the top. And one can then immediately invert this to get a way to deduce the colors of cells on a given row from the colors of certain combinations of cells in a given column.





Another consequence of additivity: the correspondence between colors of cells on rows and columns in the rule 60 cellular automaton. In each case specifying the colors of the cells that are marked with dots immediately determines the colors of the cells that are marked with diamonds. The final diamond cell is black if an odd number of the dotted cells are black, and is white otherwise. The pictures on the right show which cells in the top row and which cells in the right-hand column determine the cells at successive positions in the right-hand column and in the top row respectively. These pictures can be thought of as matrices with 1's at the position of each black dot, and 0's elsewhere. Multiplying these matrices modulo 2 by vectors corresponding to a row of the cellular automaton gives a column, and vice versa. This means that the matrix on the second row of pictures is the inverse modulo 2 of the one on the first row.

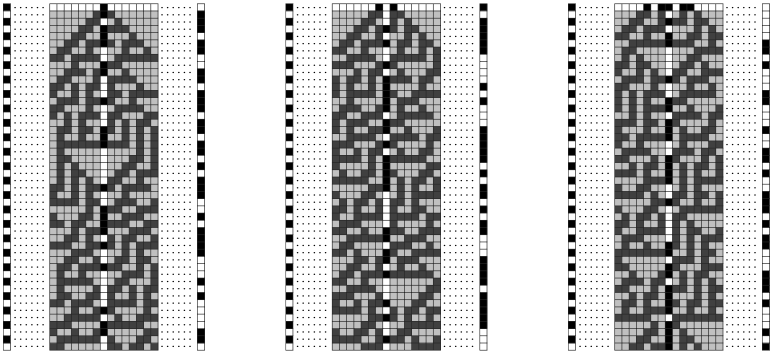
Which cells in a column are known will depend on how the encrypting sequence was formed. But with almost any scheme it will eventually be possible to determine the colors of cells at each of the positions across any register of limited width. So once again a fairly simple process is sufficient to allow the original key to be found.

So how then can one make a system that is not so vulnerable to cryptanalysis? One approach often used in practice is to form combinations of rules of the kind described above, and then to hope that the complexity of such rules will somehow have the effect of making cryptanalysis difficult.

But as we have seen many times in this book, more complicated rules do not necessarily produce behavior that is fundamentally any more complicated. And instead what we have discovered is that even among extremely simple rules there are ones which seem to yield behavior that is in a sense as complicated as anything.

So can such rules be used for cryptography? I strongly suspect that they can, and that in fact they allow one to construct systems that are at least as secure to cryptanalysis as any that are known.

The picture below shows a simple example based on the rule 30 cellular automaton that I have discussed several times before in this book. The idea is to generate an encrypting sequence by sampling the evolution of the cellular automaton, starting from initial conditions that are defined by a key.



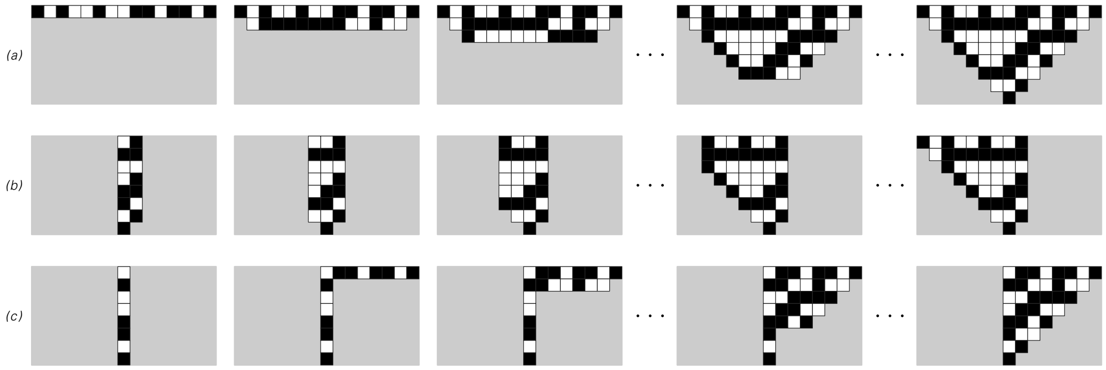
Encryption using a column of rule 30 as the encrypting sequence. I first suggested this method in 1985.

In the case of the additive cellular automaton shown on the previous page its nested structure makes it possible to recognize regularities using many of the methods of perception and analysis discussed in this chapter. But with rule 30 most sequences that are generated—even from simple initial conditions—appear completely random with respect to all of the methods of perception and analysis discussed so far.

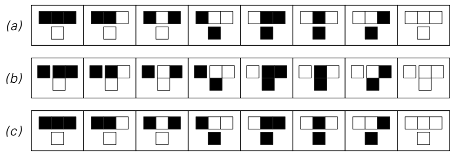
So what about cryptanalysis? Does this also fail to find regularities, or does it provide some special way—at least within the context of a setup like the one shown above—to recognize whatever regularities are necessary for one to be able to deduce the initial condition and thus determine the key?

There is one approach that will always in principle work: one can just enumerate every possible initial condition, and then see which of them yields the sequence one wants. But as the width of the cellular automaton increases, the total number of possible initial conditions rapidly becomes astronomical, and to test all of them becomes completely infeasible.

So are there other approaches that can be used? It turns out that as illustrated in the picture below rule 30 has a property somewhat like the additive cellular automaton discussed two pages ago: in addition to allowing one row to be deduced from the row above, it allows columns to be deduced from columns to their right. But unlike for the additive cellular automaton, it takes not just one column but instead two adjacent columns to make this possible.



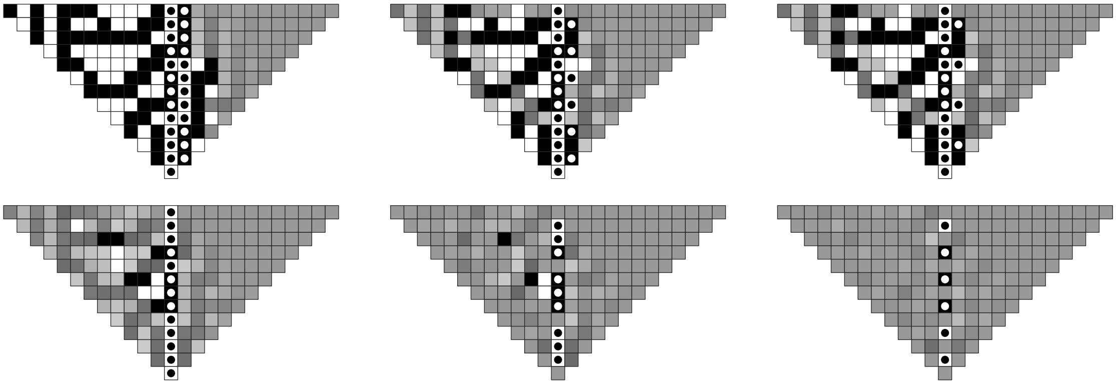
Sideways evolution in rule 30. (a) shows ordinary evolution from one row to the next. (b) shows evolution to the left starting from a pair of adjacent columns. (c) shows how a second column can be filled in from a row of cells to the right. The possibility of (b) is a consequence of one-sided additivity in rule 30; it leads to some level of cryptanalysis if the encrypting sequence consists of a complete column of cells.



So if the encrypting sequence corresponds to a single column, how can one find an adjacent column? The last row of pictures above show a way to do this. One picks some sequence of cells for the right half of the top row, then evolves down the page. And somewhat surprisingly, it turns out that given the cells in one column, there are fairly few possibilities for what the neighboring column can be. So by sampling a limited number of sequences on the top row, one can often find a second column that then allows columns to the left to be determined, and thus for a candidate key to be found.

But it is rather easy to foil this particular approach to cryptanalysis: all one need do is not sample every single cell in a given column in forming the encrypting sequence. For without every cell there does not appear to be enough information for any kind of local rule to be able to deduce one column from others.

The picture below shows evidence for this. The cells marked by dots have colors that are taken as given, and then the colors of other cells are filled in according to the average that is obtained by starting from all possible initial conditions.



Patterns generated by rule 30 after averaging over all possible initial conditions that reproduce the arrangements of colors in the cells indicated by dots. If a cell is completely black or completely white then this means that its color is uniquely determined by the constraints given. If the cell is shown as gray then this means that it has some probability of being black and some probability of being white. Note that when two complete adjacent columns are specified all the cells on the left-hand side are determined. But when fewer cells are specified, the number of cells that are determined decreases rapidly, indicating that cryptanalysis is likely to become difficult.

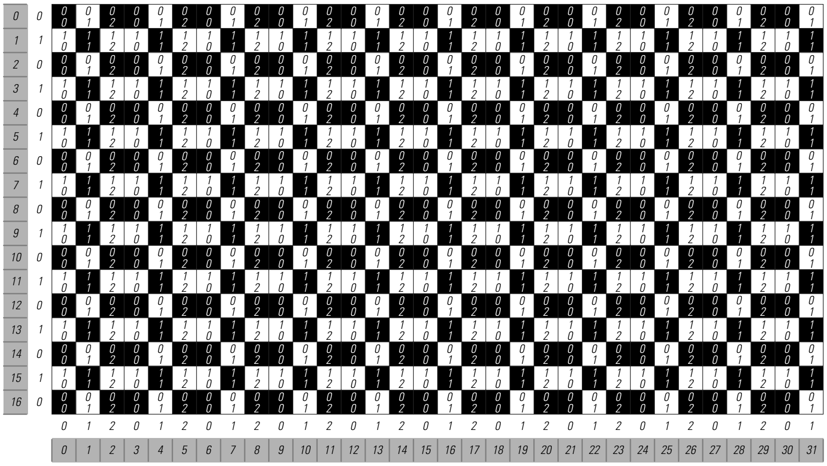
With two complete columns given, all cells to the left are determined to be either black or white. And with one complete column given, significant patches of cells still have determined colors. But if only every other cell in a column is given, almost nothing definite follows about the colors of other cells.

So what about the approach on page 602? Could this not be used here? It turns out that the approach relies crucially on the additivity of the underlying rules. And since rule 30 is not additive, it simply does not work. What happens is that the function that determines the color of a particular cell from the colors of cells in a nearby column rapidly becomes extremely complicated—so that the approach probably ends up essentially being no better than just enumerating possible initial conditions.

The conclusion therefore is that at least with standard methods of cryptanalysis—as well as a few others—there appears to be no easy way to deduce the key for rule 30 from any suitably chosen encrypting sequence. But how can one be sure that there really is absolutely no easy way to do this? In Chapter 12 I will discuss some fundamental approaches to such a question. But as a practical matter one can say that not only have direct attempts to find easy ways to deduce the key in rule 30 failed, but also—despite some considerable effort—little progress has been made in solving any of various problems that turn out to be equivalent to this one.

Traditional Mathematics and Mathematical Formulas

Traditional mathematics has for a long time been the primary method of analysis used throughout the theoretical sciences. Its goal can usually be thought of as trying to find a mathematical formula that summarizes the behavior of a system. So in a simple case if one has an array of black and white squares, what one would typically look for is a formula that takes the numbers which specify the position of a particular square and from these tells one whether the square is black or white.


0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0
0 1 2 0 1 2 0 1 2 0 1 2 0 1 2 0 1 2 0 1 2 0 1 2 0 1 2 0 1 2 0 1
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31