De Bruijn Sequences

Table of Contents

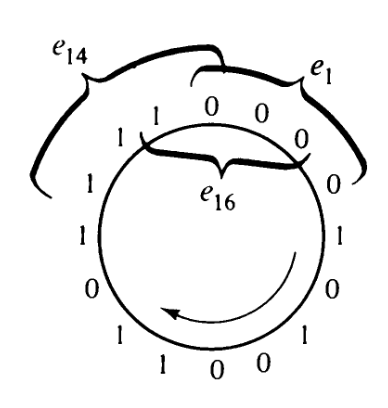
[Applications 2](#_Toc138883427)

[Brute Force Attacks 2](#_Toc138883428)

[Angle Detection 3](#_Toc138883429)

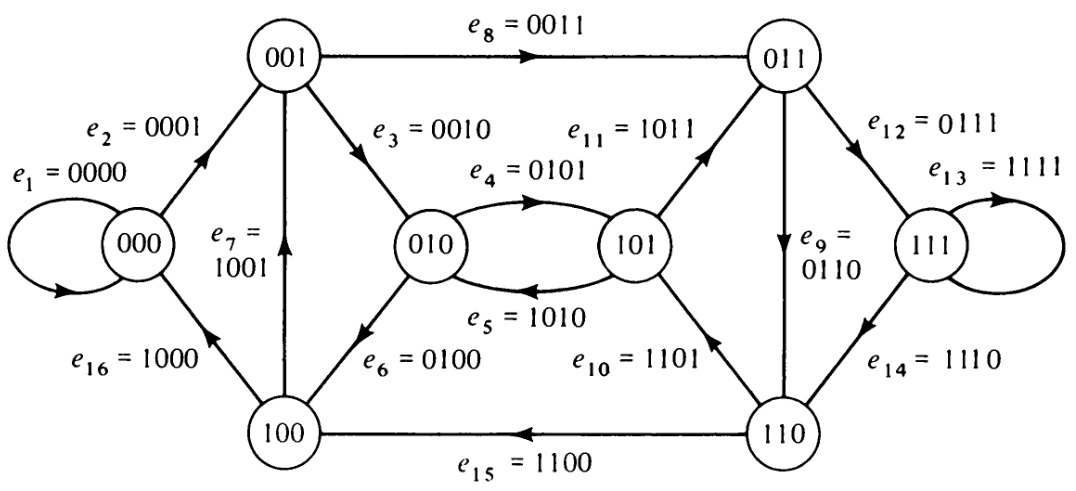
[Stream Ciphers 4](#_Toc138883430)

Suppose we have the following sequence:



If we consider groups of 4 bits, then there are 16 possible combinations. All 16 of these combinations can be formed using the sequence above. As shown by the braces, we are considering groups to be formed by shifting 1 bit at a time and also joining the end of the sequence to the start.

The sequence above is a **De Bruijn Sequence** because it is able to cover all of the combinations. We can represent this graphically. In the graph below, each node represents a 3-bit number and the edge indicates a change in which the left-most digit is dropped and the number shown on the edge is added to the right.



We can use the edges to form a sequence: 000**0**, 000**1**, 001**0**, …. Essentially, we are consider the node we are currently on and the edge used to get there (the number in bold being the edge). If we follow this sequence, we will find that it is the same as the sequence shown earlier. This sequence covers **all the edges**, thus making it an Euler line. If we change the order in which we traverse the edges while still forming an Euler line, we will end up with a different sequence that is also a De Bruijn sequence. There are possible sequences, where is the number of bits. Thus, for our case, there are possible sequences.

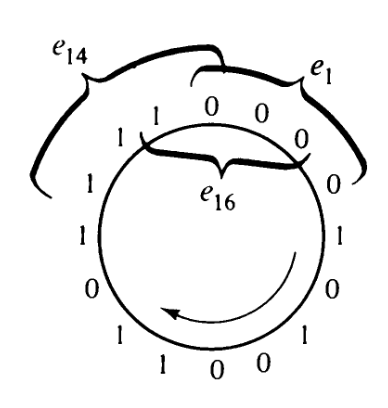
## Applications

### Brute Force Attacks

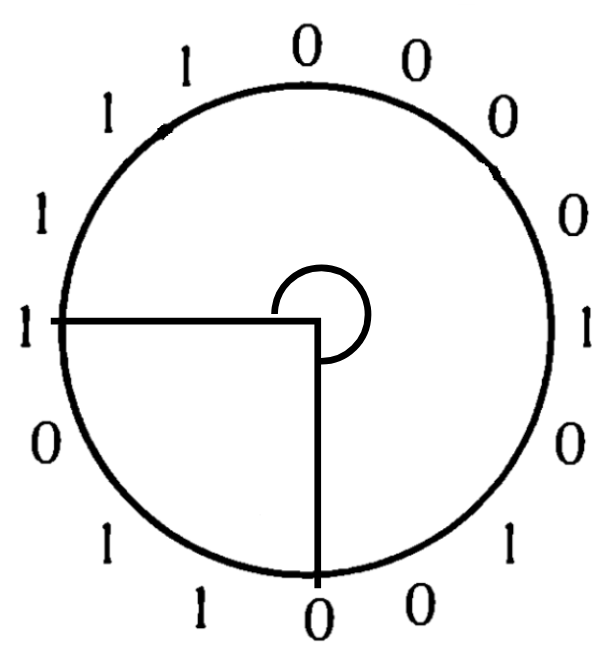
Consider a lock with a 4 digit pin. There are 10 possible digits at each position, meaning there are combinations. If we want to try all the combinations, entering each combination will require 4 button presses, giving us 40000 button presses in total. However, if we create a De Bruijn sequence and follow that, we can cover all the combinations with just 10003 button presses. For an digit pin with possible digits at each position, there are button presses.

### Angle Detection

De Bruijn cycles are also used for **angle detection** in some cases.



Suppose we choose a number, 15. The start of the sequence is always fixed at the top of the circle. Thus, the number represented by 15, 1111, starts at the left-hand side. This forms an angle, as shown below:



We will always consider the start of the angle to be at the bottom of the circle and go anti-clockwise. Thus, we can represent the angle (270 degrees) using just the number 15.

### Stream Ciphers

De Bruijn sequences are used directly in **stream ciphers**. A sequence is generated which is XORed with the bits of some plain text to create the cipher text. Since the number of possible Euler lines grows exponentially as we increase the sizes of the groups being considered in the sequence, the encryption can be difficult to crack.