Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακή Διατριβή
Στην Ασφάλεια Υπολογιστών και Δικτύων

Ανάλυση Ακολουθιών De Bruijn

.

Ανδρέας Βαρέλιας

Επιβλέπων Καθηγητής
Δρ. Κωνσταντίνος Λιμνιώτης

Μάιος 2021

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Ανάλυση ακολουθιών De Bruijn.

Ανδρέας Βαρέλιας

Επιβλέπων Καθηγητής
Δρ. Κωνσταντίνος Λιμνιώτης

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε
προς μερική εκπλήρωση των απαιτήσεων για απόκτηση

μεταπτυχιακού τίτλου σπουδών
στην Ασφάλεια Υπολογιστών και Δικτύων

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών
του Ανοικτού Πανεπιστημίου Κύπρου

Μάιος 2021

Περίληψη

Αυτή η διατριβή επικεντρώνεται στην ακολουθία De Bruijn, ως ερευνητικό θέμα αυξανόμενου ενδιαφέροντος, το οποίο «αναβιώνει» τα τελευταία χρόνια ακριβώς επειδή τα NLFSR χρησιμοποιούνται μαζικά για τη δημιουργία ισχυρών κρυπτογραφικών αλγορίθμων.

Πιο συγκεκριμένα, οι συναρτήσεις Boolean που δημιουργούν ακολουθίες De Bruijn μελετώνται σε αυτή τη διατριβή, όσον αφορά τη διερεύνηση των αντίστοιχων κρυπτογραφικών ιδιοτήτων για συναρτήσεις που δημιουργούν "παρόμοιες" ακολουθίες De Bruijn.

Συγκεκριμένα, έχοντας ως αφετηρία κάποια πρόσφατα αποτελέσματα για τον προσδιορισμό ζευγών αλληλουχιών De Bruijn [1]που μοιράζονται τη μεγαλύτερη κοινή ακολουθία, παρουσιάζουμε πρώτα έναν νέο αλγόριθμο προσέγγισης, χρησιμοποιώντας τους (αντίστροφους) πίνακες επιθήματος των ακολουθιών, για να υπολογίσουμε αποτελεσματικά ζεύγη τέτοιων De Bruijn ακολουθιών επεκτείνοντας έτσι περαιτέρω τα πρόσφατα ερευνητικά αποτελέσματα σε αυτόν τον τομέα.

Στη συνέχεια, χρησιμοποιώντας κατάλληλα εργαλεία λογισμικού, εξετάσαμε τις ιδιότητες των αντίστοιχων Boolean λειτουργιών τους, όπως αλγεβρικός βαθμός, μη γραμμικότητα και αλγεβρική ανοσία - ενώ μελετάμε επίσης πώς συμπεριφέρεται η γραμμική πολυπλοκότητα για οποιοδήποτε τέτοιο ζεύγος «παρόμοιων» ακολουθιών De Bruijn.

Δείχνουμε ότι, αν και στην πλειονότητα των περιπτώσεων αυτές οι ιδιότητες παραμένουν αμετάβλητες, ενδέχεται να έχουμε κάποιες διαφορές που θα μπορούσαν να είναι κρυπτοαναλυτικής αξίας, δημιουργώντας έτσι μια νέα ιδιότητα που πρέπει να ελεγχθεί όταν εξετάζουμε την κατασκευή γεννητριών De Bruijn.

Summary

This dissertation focuses on the De Bruijn sequence, being a research topic of increasing interest, which has been "reviving" in recent years precisely because NLFSRs are massively being used to build powerful cryptographic algorithms.

More specifically, Boolean functions generating De Bruijn sequences are studied in this thesis, in terms of investigating the corresponding cryptographic properties for functions generating ``similar'' De Bruijn sequences.

In particular, having as a starting point some recent results on identifying pairs of De Bruijn sequences sharing the longest common subsequence, we first present a new approximation algorithm, utilizing the (inverse) suffix arrays of the sequences, to efficiently compute pairs of such De Bruijn sequences, thus further extending recent research results on this field.

Subsequently, using appropriate software tools, we examined properties of their corresponding Boolean functions such as algebraic degree, nonlinearity, and algebraic immunity – whilst we also study how the linear complexity behaves for any such pair of "similar" De Bruijn sequences.

We show that, although in the majority of the cases these properties remain invariant, we may have some differences which could be of cryptanalytic value, thus establishing a new property that is of importance to be checked when we consider the construction of De Bruijn generators.

Ευχαριστίες

First, I feel obliged to express my gratitude to the:

1. Open University of Cyprus (OUC) that provided me the opportunity to attend a postgraduate course at my age simply and efficiently,

2. Academic staff for their passion, professionalism, and high educational standard,

3. Administrative staff that made this process worked.

Secondly, to my family (wife, children, and grandchildren) and friends for their encouragement and trust /confidence in taking this course.

Finally, my supervisor Dr. Constantinos Limniotis for his continued support and patience that contributed the utmost to complete this dissertation.

Contents

Chapter1

Introduction

1. Introduction:

De Bruijn binary sequences are an important family of mathematical structures, sequences, with many applications, including cryptography. These are 2^n period binary sequences, where each possible n-tuples of bits appear exactly once within the sequence.

In cryptographic applications, and especially in stream ciphers, these are sequences generated by non-linear registers (NLFSRs) of size n, which during their operation "run" all possible states (maximal length NLFSRs), thus producing sequences with the maximum possible period. The output of such full-cycle NLFSRs is De Bruijn sequences. There are several known constructions of De Bruijn sequences, most of which are based on the formation of n-order sequences beginning with lower-order sequences, of which many other primary constructions are also known. But these known constructions cannot completely cover the space of the De Bruijn binary sequences which have a range of $2^{2^{n-1}-n}$ that is, as well as the Hamiltonian paths in a De Bruijn graph.

Therefore, the problem of describing a method capable of generating any possible De Bruijn sequence remains open.
Despite many years of a thorough study of the De Bruijn sequences, there are still several open questions. First, new techniques for constructing De Bruijn sequences are constantly appearing in the literature. Also, very recently (2020), a new research result presents a property that meets two De Bruijn sequences that have one subdivision in common, while at the same time there is no other De Bruijn sequence with a longer common subdivision

1.1. Scope of the study

This dissertation will focus on De Bruijn sequences, emphasizing on further promoting some recent results on the field. More precisely, in very recent research work [1], the longest subsequences shared by two de Bruijn sequences are being investigated. The researchers therein proved that, for any fixed de Bruijn sequence, we can find another De Bruijn sequence sharing the longest subsequence with the original one by a single cross join operation from it. Then determining such sequences is equivalent to finding cross-join pairs with maximum diameter. However, they do not present any specific algorithm to find such a solution to this problem – i.e. to efficiently find cross-join pairs with the maximum diameter. In this dissertation, we focus on further promoting these results, through utilizing the so-called suffix arrays to facilitate the computation of cross-join pairs with the maximum diameter.
The motivation for this research approach is the fact that the so-called "cross-join pairs" technique is strongly associated with the notion of constructing De Bruijn sequences via their suffix arrays, as it has been also recently shown in 2018 [2].
Therefore, both a theoretical foundation of the correlation of these arrays with the above property, as well as the practical confirmation through the implementation of an appropriate testing environment implementing a novel approximation algorithm will be attempted.
Furthermore, for each such pair of sequences, an attempt will be made – through a test environment to:

- Derive the corresponding subsequence that shares the longest common subsequence,
- Investigate the corresponding cryptographic properties of the corresponding Boolean functions that produce them,
    - Assess whether the results can be used in the production of stronger NLSFR's with the least effort.

1.2. Key Research Questions

i. Given a binary De Bruijn sequence y, find an algorithm to compute another binary De Bruijn sequence y ’with the largest common subsequence, utilizing the so-called inverse suffix arrays

ii. Which are the relationships between the cryptographic properties of two Boolean functions generating “similar” De Bruijn sequences, in terms of the above construction?

iii. Establish the importance of the construction of NLSFR’s using generators with stronger cryptographic properties with minimum effort and resources.

1.3. Objectives of the study

Although it is not explicitly mentioned [1], the problem of finding pairs of De Bruijn sequences sharing the longest common subsequence is of high cryptographic importance; More precisely, for a given cryptographic sequence, finding another one that resembles the initial one but it has cryptographic weaknesses may be the starting point for mounting successful cryptanalytic attacks – for example, linear or low order approximation attacks. Such cryptographic weaknesses are in turn strongly related to the cryptographic properties of the Boolean functions that generate the corresponding sequences. Therefore, a research question that naturally appears is the following: *given two De Bruijn sequences *s* and *s’* of the same order n with a common subsequence *y* such that there is no other De Bruijn sequence *s”* sharing with *s* a larger common subsequence than *y*, what is the relationship between the cryptographic properties (i.e. algebraic degree, nonlinearity, algebraic immunity, etc.) of the two corresponding Boolean functions that generate *s* and *s’* respectively? Bearing in mind that De Bruijn sequences are generated by full-cycle Nonlinear Feedback Shift Registers (NLFSRs) which in turn are being used in contemporary lightweight stream ciphers, it becomes evident that this thesis focuses on a topic of increasing interest.

To this effect, sequences derived from randomly constructed ones with the cross join pair operation will be examined focusing on the concept of whether nearby sequences formed by longest sharing common subsequences present improved cryptographic properties that can be utilized as generators of powerful NLFSR's with minimum effort.

1.4. Methodology

The method of the research will be:

- mathematical, in the sense that an attempt will be made to highlight properties which will describe how, from the suffix arrays of two De Bruijn sequences, the larger common parts of these sequences can be easily found,
- experimental, in the sense that a computational platform will be developed to test a large sample adequate to enable drawing results and solid conclusions.

The experimental part is expected to initially help in the emergence of properties (which will then be proved mathematically) and, subsequently, to confirm the grounded theoretical result.

Furthermore, to find the cryptographic properties of the Boolean functions that produce De Bruijn sequences, the Sage software will be utilized, while a special Boolean function finding algorithm is developed.

1.5. Structure of the Dissertation

This thesis is structured as follows:

Chapter 1: Introduces the concept of the core of the study and defines its scope. It specifies the key research questions and formulates the goals to address those. Finally, it described the methodology to be followed for achieving them.

Chapter 2: Provides background information about De Bruijn and its discovery both in general terms and their binary form.

Chapter 3: Deals with the construction of primary De Bruijn sequences and provides a brief description of the process used.

Chapter 4: Refers to the cryptographic criteria that will be used in the assessment in the analysis of the primary and the derived sequences.

Chapter 5: Provides the methodology on how to derive De Bruijn sequences using the cross-join pair operation and their relationship with the primary as far as the shared part of the two sequences.

Chapter 6: Presents the experimental part of the exercise e.g. the creation of the primary and the cross pair join operation to derive the new sequences and their conditioning to produce the corresponding Boolean function that is required for part of their cryptographic criteria.

Chapter 7: Tabulates the experimental results found in chapter 6.

Chapter 8: Discusses the findings, appends the conclusions drown and describes the way forward.

Chapter 2

Background Information

2. Who is De Bruijn

Nicolaas Govert (Dick) de Bruijnn was the full name of the Dutch mathematician born on July 7th, 1918 in Hague who greatly contributed to many fields of mathematics. In particular, he dealt with analysis, number theory, combinatorics, and logic, and many others.
However, he became very well known for his work in the discovery of the sequence that was named after him e.g. De Bruijn Sequences.
De Bruijn was triggered by long-standing efforts commenced as early as the 3rd – 2nd century BC to formulate ways of finding mnemonics remembering long names or sentences and in 1946 he presented his work where sequences are optimally short concerning the property of containing every string of length n at least once.
Based on this property, several applications were developed including cryptography, biology, neuroscience, psychology, text analysis, and many others where combinatorics are of primary concerns in their study and support.
De Bruijn during his lifetime he contributed to many other topics of mathematics like:

- discovering an algebraic theory of the Penrose tiling and, more generally, discovering the "projection" and "multigrid" methods for constructing quasi-periodic tilings,
- the De Bruijn–Newman constant,
- the *De Bruijn–Erdős theorem*, in graph theory,
- a different theorem of the same name: the *De Bruijn–Erdős theorem*, in incidence geometry,
- the BEST theorem in graph theory,
- De Bruijn indices.

Also, he wrote one of the standard books in advanced asymptotic analysis (De Bruijn, 1958).

In the late sixties, he designed the Automath language for representing mathematical proofs, so that they could be verified automatically (see automated theorem checking). Shortly before his death, he had been working on models for the human brain.
De Bruijn, died on February 18th, 2012 at the age of 94.

2.1 De Bruijn sequences

A De Bruijn sequence in its general form is a cyclic structure of alphabet *A* with k elements and order *n* which denotes every possible length-*n* string on *A* that occurs exactly once as a substring e.g. as a contiguous subsequence.
Such a sequence is denoted by $B(k, n)$ and has length k^n, the combinations k elements arranged in groups n elements which is also the number of distinct strings of length *n* on *A*.
Each of these distinct strings, when taken as a substring of $B(k, n)$, must start at a different position, because substrings starting at the same position are not distinct.
Therefore, $B(k, n)$ must have *at least* k^n symbols, and since $B(k, n)$ has *exactly* k^n symbols, De Bruijn sequences are optimally shorter taking into account the restriction of containing every string of length *n* at least once.
In this respect, the number of distinct de Bruijn sequences $B(k, n)$ is:

$$N = [k(!))^{k^{n-1}}]/k^n$$ [3]

The above is valid for the general case where A with k element ∈ [a,b,c….z] and the size of the grouping is a positive integer n.
The functionality of the De Bruijn sequences is appreciated in the below example:
Consider that we have an electronic lock that uses a 4-digit combination as a key.
In short, to open it we have 10,000 combinations in groups of 4 digits in a brute force attack operation. This will require 40,000 keystrokes until the correct combination is reached.
However, if this is replaced by a way to run every four digits of a De Bruijn sequence and each time only the last four digits with overlap are entered then the process is shorter to 10,003 keystroke entries.
Should the above be confined in the A ∈ [0,1] form it defaults to a binary sequence with application in cryptography.

2.2 Binary De Bruijn sequences

Binary De Bruijn sequences are a subset of the general form where the alphabetic elements are limited to 0 and 1 e.g. A [0,1] and k =2.

Therefore, the general formula for the total number of distinct De Bruijn sequences with length 2^n is reduced to $N = 2^{2^{n-1}-n}$

This type of sequence is forming the base for the construction of NLFSR's and due to their uniqueness, length and complexity are particularly useful in encoding messages.

De Bruijn sequences are balanced and are in line with several other pseudorandom properties. Therefore, any maximum period FSR generates a De Bruijn sequence. It is easy to see that it is the feedback function of an FSR that generates a De Bruijn sequence, it holds $h(0, 0, \ldots, 0) = 1$ (to avoid the all-zeros cycle) and $h(1, 1, \ldots, 1) = 0$ (to avoid the all-ones cycle).

Choosing through an NLFSR that generates the De Bruijn sequence is still an open problem since only a few such feedback function families are known.

This thesis focuses explicitly on binary De Bruijn sequences.

Chapter 3

Primary Construction of Binary De Bruijn sequences

3 Construction of De Bruijn Sequences [4]:

This chapter studies the methods for constructing binary De Bruijn sequences from scratch e..g. just by defining the degree n of the prospective sequence.

There are several methods for constructing primary sequences and below four of the most representative are presented.

This particular study, however, will use only the method based on the Suffix Array with the aid of a c-based program.

3.1. De Bruijn Graph,

It is considered the most representative method for constructing De Bruijn sequences and is implemented with the aid of graphs which in this particular case are called De Bruijn Graphs.

The process for an n degree sequence requires the construction of a graph with 2^n nodes. Each node accommodates an n bit subsequence involved in the construction of the sequence.

Each node is connected with the next in a sliding manner so that the next there must be an overlap of connected nodes in (n-1) bits of the subsequence of each node.

In this way, a directed graph is constructed with all possible links that are satisfying the above definition.

The next step is to choose the nodes that are connected in such a way where the sequence that is formed from the bits when the corresponding subsequences are placed in an n-1 overlap manner appears only once.

The product of the above operation is a 2^n De Bruijn sequence.

More pictorially, this can be represented by an n=3 example.

For n=3, a graph with 8 nodes will be created (Fig.3.1).

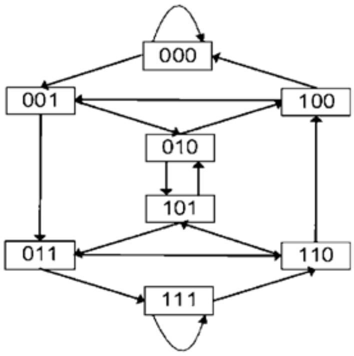


Fig. 3.1. De Bruijn graph of an n=3 sequence [2]

In the above Graph, two nodes are connected ($A \rightarrow B$), only if the last 2 bits of the sequence of node A are the same as the first 2 bits of the sequence of node B. In general terms, two nodes are connected, only if the last (n-1) bits of the node A sequence are the same as the first (n-1) bits of the node sequence of node B.

Following the route: 000 → 001 → 010 → 101 → 011 → 111 → 110 → 100, you will pass exactly once from each sequence of size n = 3 and we will create the De Bruijn sequence (00010111), as shown below with overlaps:

0	**0**	**0**	**1**	**0**	**1**	**1**	**1**	**(0**	**0)**
	0	0	1						
		0	1	0					
			1	0	1				
				0	1	1			
					1	1	1		
						1	1	0	
							1	0	0

Fig. 3.2. Table showing the De Bruijn shifting and overlapping.

It must be noted that, to produce a De Bruijn sequence all the nodes should be included. Also, the same can be constructed by joining individual circles which despite that they are following the rule of the overlap they do not traverse through all the nodes.

In this case, individual cycles could be joined together by connecting them at their conjugate terms.

3.2. Necklace (prefer -one) method [1]

This is the simplest method for constructing De Bruijn sequences. However, its effectiveness is limited to small n due to the high computational requirements as the n increases.

It starts with n zeros and adds bit 1 to the sequence whenever the string of bits is not repeated otherwise bit 0 is added.

Example for n=4.

- Starts with 0000 and adds 1,1,1,1,. Adding another 1 will create a string of 11111 which comprises two tuples of 1111 then a 0 is added.
- Then add 1 and check if the newly generated tuple is repeated. If yes, then add 0 otherwise continue with 1's.

Finally, the method generates the sequence 0000111101100101(000).

Prefer-same and prefer-opposite methods are similar to the prefer-one method using different bit insertion criteria. Given the same initial state, these methods generate only one de Bruijn sequence.

3.3. Method of Arithmetic Remainders

A De Bruijn sequence can be constructed using the technique of the arithmetic remainders and is applied only in the case of the binary one.

Considering a sequence of degree n and initial value $a_1=2^n-1$ a De Bruijn sequence can be constructed by repeated replacements of the number produced by the function $a_{i+1}=2a_i \pmod{2^n}$

for the first n numbers.

In case, however, where $i \leq j$, $a_i = 2a_j$ is valid then the formula is given by the relation: $a_{i+1}=2a_i +1 \pmod{2^n}$.

This practically means that if the outcome of the first function is a number that has already been generated, then we add a_1 to it to obtain the new number and to continue to the stage.

Finally, the numbers that are generated are converted into their binary form (with appropriate completion zeros where required) and combined to produce the De Bruijn sequence.

To make it clearer, a numerical example is appended here below for n=3 and initial value $a_1 = 2^3 - 1 = 7$:

Function:	**Binary form:**
$a_1 = 2^n - 1(\text{mod}2^n) = 2^3 - 1\text{mod}(8) = 7$	111
$a_2 = 2\, a_1\, (\text{mod}2^n) = 14 \text{ mod}(8) = 6$	110
$a_3 = 2\, a_2\, (\text{mod}2^n) = 12 \text{ mod}(8) = 4$	100
$a_4 = 2\, a_3\, (\text{mod}2^n) = 8 \text{ mod}(8) = 0$	000
$a_5 = 2\, a_4\, (\text{mod}2^n) = 0 \text{ mod}(8) = 0$ then $a_5 = 2\, a_4 + 1(\text{mod}2^n) = 1 \text{ mod}(8) = 1$	001
$a_6 = 2\, a_5\, (\text{mod}2^n) = 2 \text{ mod}(8) = 2$	010
$a_7 = 2\, a_6\, (\text{mod}2^n) = 4 \text{ mod}(8) = 4$ then $a_7 = 2\, a_6 + 1(\text{mod}2^n) = 5\text{mod}(8) = 5$	101
$a_8 = 2\, a_7(\text{mod}2^n) = 10 \text{ mod}(8) = 2$ then $a_8 = 2\, a_7 + 1(\text{mod}2^n) = 11 \text{ mod}(8) = 3$	011

The process is now completed and the binary configuration of the $a_1 \ldots a_8$ are placed a chain manner with the a_{i+1} overlays the last n-1 elements of a_i.

In doing so, the sequence 11100010(00) is produced which is an n=3 De Bruijn sequence [5].

3.4 Suffix Arrays. [2]

The methods described earlier are approaching the construction of the De Bruijn sequence in a bit-by-bit manner, while the new technique puts all the possible n-tuples in an order to produce the Suffix Array S step by step.

Suffix Array in general is a data structure that is made to holds all the elements of a data string in a sorted manner.

In this case Suffix Array S of a binary sequence yN holds the starting positions (ranging from 0 to N – 1) of its N lexicographically ordered suffixes; i.e. for $0 \leq i < j < N$, it holds $y_{S[i]}^{N-1} < y_{S[j]}^{N-1}$ [6]

To achieve this, it is required to set the parameters and follow properties as described below:

Set **k** and **m** such that **S[k] = m** for any **0 ≤ k, m < 2n**.

Setting **k** and **m** as in (1) then $\mathbf{y_m^{m+n-1}}$ coincide with $\mathbf{k^{(n)}}$**.**

Property 1: If y is a binary De Bruijn sequence of order n and S is its corresponding suffix array, then the following conditions are valid:

$\mathbf{S[1] = S[0] + 1, S[2^{2n-1}] = S[0] - 1,}$ [7]

$\mathbf{S[2^n - 2] = S[2^n - 1] + 1, S[2^{n-1} - 1] = S[2^n - 1] - 1,}$

$\mathbf{S[2^n - 1] - S[0] \geq n,}$

$\mathbf{y_i = 0}$ **if and only if** $\mathbf{S^{-1}[i] < 2^{n-1}}$ **.**

where, all operations are performed **mod** $\mathbf{2^n}$.

The process starts by initializing the first *n* bits of the *n*-th order De Bruijn sequence to zero. With $\mathbf{y_0^{n-1}}$ set to 0, then by the conditions of property 1(i) the **S[1] = S[0] + 1, S[22n-1] = S[0]– 1.**

The main function from hereon is to assign m to an S(k) in line with the conditions counted in 2(i) - 2(iv).

To this effect, there are two possible options for **k.** Should S(k) not be assigned yet **k** is chosen such that S(k)=m, f=1.

Otherwise, if none of the k is available, then f=0, and the whole process revisits the previous assignments this time starting from S^{-1}[m-1].

Reassignment of S^{-1}[m] however, to S^{-1}[m-1] leaves only one option as the former was deemed in conflict and is selected.

In the case where both options are found in conflict previous assignments are reexamined.

More practically, an example is appended here below appended providing a vivid description of the process.

Example:

For n=3 the size of the sequence is N= 2^3 =8.

Assuming the final sequence shall be of the form 0****111.

Considering that tuple 111 is at the end of the sequence then the position after wrapping is 0.

To this effect, for k=0 the following are determined.

Tuple 111 will start at position 5 e.g. S[8]=5, S[1]=0 or 2 depending on whether wrapping is considered or not.

Consequently, suffix111 should be followed by 0 and preceded with 0. Hence position 1 must accommodate the suffix 000, position 6 must accommodate the suffix 110, and position 7 the suffix 100.

Also, Suffix 001 will follow 000 in position 2.

So, the table after initialization has the below configuration:

Serial No	Position	Suffix
1	1	000
2	2	001
3	*	010
4	4	011
5	7	100
6	*	101
7	6	110
8	5	111

Fig.3.3 Suffix Array of De Bruijn sequence for k=0 in construction.

So, what's left is the determination of positions 3 and 6 for suffixes 3 and 5.

By default, the 001 at position 2 can only be followed 01*. Therefore, only the terms 010 and 011 fit the situation and 010 is chosen. The 011 will be accommodated last and the array is taking the shape as below:

Serial No	Position	Suffix
1	0	000
2	1	001
3	2	010
4	4	011
5	7	100
6	3	101
7	6	110
8	5	111

Fig.3.4 Final Suffix Array of De Bruijn sequence for k=0.

In the case of **k=1** the array is shaped as:

Serial No	Position	Suffix
1	1	000
2	2	001
3	*	010
4	4	011
5	7	100
6	*	101
7	6	110
8	5	111

Fig.3.5 Suffix Array of De Bruijn sequence for k=1 in construction.

In the case of **k=2** the array is shaped as:

Serial No	Position	Suffix
1	2	000
2	3	001
3	*	010
4	4	011
5	*	100
6	*	101
7	6	110
8	5	111

Fig.3.6 Suffix Array of De Bruijn sequence for k=6 in construction.

Chapter 4

Cryptographic Criteria

Cryptographic Criteria:

As mentioned earlier, the De Bruijn sequences are involved in the support of several sciences. In this study, however, the main objective is to look into these sequences for their cryptography support and their crypto properties are of utmost importance in the analysis of the relationship between the primary (constructed) and secondary (derived) ones.

These properties are Linear Complexity as applied directly to the sequence and the Algebraic degree, Non-Linearity, and Algebraic Immunity as applied to the Boolean function that produces the sequence.

Indeed, the linear complexity is an important cryptographic property for any cryptographic sequence and it needs to be high (as described next). On the other side, to generate sequences with large linear complexity, nonlinear Boolean functions (as described next) are employed. However, even if the generated sequence has high linear complexity, there may be weaknesses in the cryptographic system if the Boolean function does not satisfy specific cryptographic properties. All these are being discussed next.

It should be pointed out that, for this reason, in this thesis, the sequences under examination will be converted to the corresponding Boolean function that produces them.

4.1 Linear Complexity: [8]

The linear complexity of a sequence is defined as the degree of the shortest linear recursion which generates the sequence. This ranges, for the specific case of De Bruijn sequences, from $2^{n-1}+n$ (lower end) to 2^n-1 (higher end).

High linear-complexity L provides strength to the sequence and subsequently to the generator that it supports since is a primary function against the Berlekamp-Massey algorithm where if 2L consecutive bits of the sequence is known, it suffices to compute the remaining bits.

Additionally, high linear complexity resists the fast algebraic attacks where some Boolean functions are not in a position to support.

4.2 Boolean Function. [7]

A Boolean function with n variables is a function with n binary inputs and one binary output. Therefore, the feedback function of an NLFSR is a Boolean function.

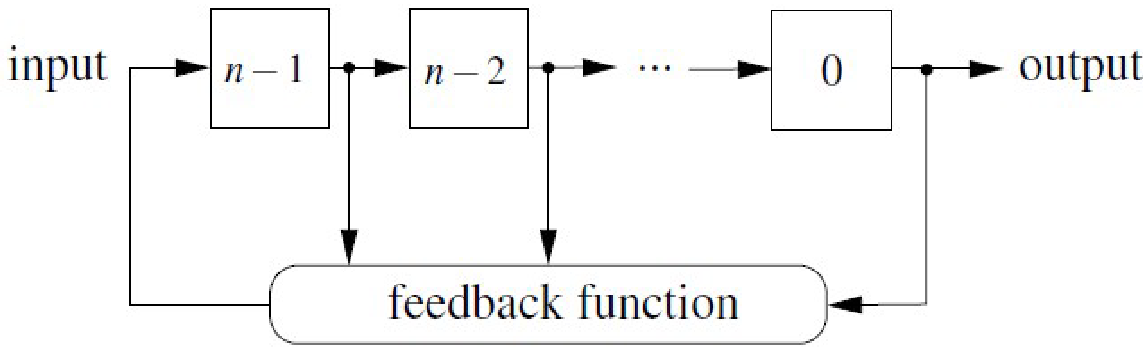


Fig.4.1 A typical diagram of an NLFSR with n position.

Consequently, the corresponding function generating – as a feedback function of an NLFSR – a sequence is determined by:

- Writing the truth table in the form of X_n,, X_{n-1}, X_{n-2} …X_0..
- Adding the corresponding output e.g., next bit.
- The vector produced at the output is the Boolean function.

This is more easily understood using the following example for a De Bruijn sequence:

Let us consider the De Bruijn sequence, **0 0 0 0 1 1 1 1 0 1 1 0 0 1 0 1** which can be expanded as follows:

De Bruijn sequence, **0 0 0 0 1 1 1 1 0 1 1 0 0 1 0 1** is expanded as follows:

X3	X2	X1	X0	NXT Bit
0	0	0	0	1
0	0	0	1	1
0	0	1	1	1
0	1	1	1	1
1	1	1	1	0
1	1	1	0	1
1	1	0	1	1
1	0	1	1	0
0	1	1	0	0
1	1	0	0	1
1	0	0	1	0
0	0	1	0	1
0	1	0	1	0
1	0	1	0	0
0	1	0	0	0
1	0	0	0	0

Fig.4.2 Binary presentation of a De Bruijn sequence in a sequential format.

The table of the sequence to be converted accommodating the Boolean function that produces it Must have the format is accepted by the Sagemath 9.2 tool.:

X0	X1	X2	X3	OUTPUT
0	0	0	0	
1	0	0	0	
0	1	0	0	
1	1	0	0	
0	0	1	0	
1	0	1	0	
0	1	1	0	
1	1	1	0	
0	0	0	1	
1	0	0	1	
0	1	0	1	
1	1	0	1	
0	0	1	1	
1	0	1	1	
0	1	1	1	
1	1	1	1	

Fig.4.3 Binary presentation of a Boolean Function format.

Sorting the initial table to take the form of the Boolean function results to:

X0	X1	X2	x3	Nxt Bit
0	0	0	0	1
1	0	0	0	0
0	1	0	0	0
1	1	0	0	1
0	0	1	0	1
1	0	1	0	0
0	1	1	0	0
1	1	1	0	1
0	0	0	1	1
1	0	0	1	0
0	1	0	1	0
1	1	0	1	1
0	0	1	1	1
1	0	1	1	0
0	1	1	1	1
1	1	1	1	0

Fig.4.4 Binary presentation of a De Bruijn sequence in a Boolean Function format that produces it.

The Boolean function (actually the output in its truth table) the produces the De Bruijn sequence 0 **0 0 0 1 1 1 1 0 1 1 0 0 1 0 1** is:

1 0 0 1 1 0 0 1 1 0 0 1 1 0 1 0

The cryptographic usefulness of Boolean functions is measured by some cryptographic characteristics which are indicative for preventing cryptanalytic attacks exploiting linear approximation and differential characteristics. The most important of these properties are:

- balancedness,
- algebraic degree,
- nonlinearity, resiliency,
- algebraic immunity.

4.2.1 Balancedness,

The balancedness relates to having an equal number of "0" and "1" in the output column of the truth table. Clearly, a De Bruijn sequence is balanced and, thus, the corresponding Boolean function is also balanced. Therefore, there is no need to study balancedness and we focus on the remaining three criteria.

4.2.2 Algebraic Degree,

The algebraic degree, deg(f), is the number of variables in the highest order term with non-zero coefficients.

Boolean functions of high degree make the attack based on Berlekamp - Massey algorithm less effective

4.2.3 Non Linearity,

The nonlinearity of a Boolean function is defined as the min of the hamming distance of the function from all the linear functions in the set.

Functions with high nonlinearity resist fast-correlation attacks as well as linear approximation attacks.

4.2.4 Algebraic Immunity (AI).

The algebraic immunity AIn(f) of an n-variable Boolean function f is defined to be the lowest degree of nonzero functions g such that fg = 0 or (f + 1)g = 0.

The algebraic immunity is an indicator of the resistance to algebraic attacks for a given Boolean function that produces De Bruijn sequences. Hence they should have high algebraic immunity to avoid low degree multivariate relations between key and output of boolean function since low degree system of equations can be easily solved.

Chapter 5

A new algorithm for finding De Bruijn sequence sharing the LCS [8]with an initial one.

This chapter will deal with the derivation of new De Bruijn sequences from existing constructions using the Cross-join Pair Operation with particular attention to maximizing the commonly shared subsequences between the two (primary and the derived one).

For this reason, the relevant theorems, definitions, and proposition will be appended to support the method:

Theorem 1: Let $f(x_0, x_1, \ldots, x_{n-1}) = X_0 \oplus f1(x_1, x_2, \ldots, x_{n-1})$ be the feedback function of a nonlinear feedback shift register. Then the feedback function $g = f \oplus I\ (a_1, a_2, \ldots, a_{n-1})$ joins the two cycles together if the conjugate pair $(0, a_1, \ldots, a_{n-1})$ and $(1, a_1, \ldots, a_{n-1})$ are in different cycles of f and splits the cycle into two if the conjugate pair is in a same cycle of f. [1]

Definition 1: Let $a = (a_0, a_1, \ldots, a_{T-1}, \ldots)$ be a sequence with period T . If the states ai , a j , ak and al with $0 \le i < j < k < l \le T - 1$ satisfy that $a_k = \overline{a}_i$ and $a_l = \overline{a}_j$ ($\overline{a}$ is the corresponding conjugate), then they are called a cross-join pair of the cycle corresponding to a, denoted by [i , j , k, l]. [1]

Theorem 2: Let a and b be two order n de Bruijn sequences. Then a can be transformed into b by repeatedly using cross-join operations. [1]

Definition 2: A cross-join pair cuts a cycle to four subsequences. The maximum length of the four subsequences is defined to be the diameter of the cross-join pair. [1]

Proposition 3: Let a be a de Bruijn sequence of order n. Assume it has a cross-join pair with diameter d, and b is the de Bruijn sequence generated by the cross-join operation. Then among the four subsequences cut by the cross-join pair, each subsequence of length d appended the next n − 1 bit is the longest subsequence shared by a and b. The length is d + n − 1. [1]

Theorem 4: Let a be a de Bruijn sequence of order n. Assume b is a de Bruijn sequence sharing the longest subsequence with a. Then b is generated from a by a single cross-join operation. [1]

Proposition 5: For $n \geq 3$, let a be a de Bruijn sequence of order n. Then there exists a de Bruijn sequence of order n sharing a subsequence of length at least $2^{n-1} + n - 3$ with it. [1]

Proposition 6: For $n \geq 3$, there is a de Bruijn sequence sharing a subsequence of length $2n - 3$ with prefer-one sequence.

Proposition 7: If [0, i , j , n+1] is a cross-join pair of some de Bruijn sequence, then $i + j \neq n + 1$. [1]

Proposition 8: If [0, i , j , n + 1] is a cross-join pair of some de Bruijn sequence, then gcd(i , j , n + 1) = 1. Proposition 7: If [0, i , j , n+1] is a cross-join pair of some de Bruijn sequence, then $i + j \neq n + 1$.

Theorem 9: For n = 3 and 4, there do not exist two de Bruijn sequences with order n sharing a subsequence of length $2^n - 2$. [1]

Theorem 10: For $n \geq 5$, there exist two de Bruijn sequences with order n sharing a subsequence of length $2^n - 2$. [1]

To this effect, we subsequently propose a new approximate that aims to find a cross-join pair with a large (possibly the largest) diameter; the main idea is to first find out a conjugate pair with the smallest possible distance d1 between the corresponding indices (see Line 1 of the Algorithm) and, next, to find out another conjugate pair to constitute (with the initial one) a cross-join pair and this new pair has the smallest possible distance d2 between its indices (see Lines 2-6 of the Algorithm 1). Of course, this is an approximation algorithm since the derived solution is not always optimal. However, for specific cases, we may be sure that the derived solution is optimal (e.g. if $d2 \leq d1 + 1$), whilst in any case, it suffices to provide a "good" solution efficiently.

Algorithm 1: Find the cross-join pair with the (approximately) largest diameter in a De Bruijn sequence.
Input: Inverse suffix array S^{-1}, of a De Bruijn sequence s of and order n
Initialization: $d \leftarrow 2^n$
1: $\{i, j\} \triangleq \min_{0 \leq i < j < 2^n - 1} (j-i) : |S^{-1}[i] - S^{-1}[j] = 2^{n-1}, j > i+1$
2: For $k \leftarrow i+1 : j-1$ *do*
3: Find $\ell : |S^{-1}[\ell] - S^{-1}[k] = 2^{n-1}$
4: if $\ell - k < d$ then
5: $d \leftarrow \ell - k$
6: end if
7: end for
Output: I, k, j, ℓ being indices of a cross-join pair with a large diameter.

The above new algorithm aims at the derivation of De Bruijn sequences from existing ones with a particular objective to maximize the length of the subsequences that is common to both the primary and the derived sequence. According to theorem 9, sequences of n<5 do not have De Bruijn sequences which are sharing a subsequence of 2^n-2. Since the objective is to derive sequences with the longest common shared subsequences then our algorithm considers only sequences of n>5. Note that this algorithm utilizes the inverse suffix array of the sequence. More precisely, the primary sequence is presented in its Inverse Suffix Array form (ISA). The main property that we observed, based on the results from is the following:

Proposition: Let *s* be a binary De Bruijn sequence of order n and S^{-1} its inverse suffix array. Then, of any 0≤ i ≤ 2^n-1$, it holds S^{-1}[i]=k, where k is the integer, whose binary representation is $s_i s_{i+1} \ldots s_{i+n-1}$.

Example: Let us consider the De Bruijn sequence s=0000101001101111. As it is shown in [2], its suffix array is S=(0 1 2 7 5 3 8 11 15 6 4 10 14 9 13 12), whilst its inverse suffix array is S^{-1}=(0 1 2 5 10 4 9 3 6 13 11 7 15 14 12 8):. By observing S and S^{-1}, the property implied by the above Proposition is easily seen (i.e. 0000 ->, 0001 -> 1, 0010 -> 2, 0101 -> 5 etc.).
Note also that the indices of a conjugate pair in the inverse suffix array have a difference of 2^{n-1} – i.e. in the above examples, the conjugate pairs are indexed as follows: (0,8), (1,9), (2,10), (3,11), (4,12), (5,13), (6,14), (7,15).

5.1 Cross-Joint Pair Operation, [9]

The ISA is placed as a vector column in an Excel spreadsheet.

Based on the fact that the conjugate elements of the ISA have a difference of x=$2^{n-1,}$ a new vector is created by taking the mod of the ISA (**=MOD(X,16**) elements with base x.

In this way, the new vector will comprise elements that are repeated/duplicated.

Each duplicated pair form a conjugate pair.

The ISA, the conjugate, and the serial no vectors for an n=5 De Bruijn sequence are shown in fig. 5.1 below.

Serial No	CONJUC.	ISA
1	0	0
2	1	1
3	3	3
4	7	7
5	15	15
6	15	31
7	14	30
8	12	28
9	9	25
10	3	19
11	6	6
12	12	12
13	8	24
14	1	17
15	2	2
16	5	5
17	10	10
18	5	21
19	11	11
20	7	23
21	14	14
22	13	29
23	11	27
24	6	22
25	13	13
26	10	26
27	4	20
28	9	9
29	2	18
30	4	4
31	8	8
32	0	16

Fig. 5.1. ISA and conjugate vector of a random n=5 De Bruijn sequence.

The next action is to find the distance between the conjugate pairs and create a third vector. One way to achieve this distance information it could be done by:

- In parallel to the ISA, conjugate a third vector is added indicating the serial number of the ISA element in the array.
- The total matrix is sorted by the conjugate vector so that the conjugated pairs are in couples and right next is the vector with the initial serial number (Fig.5.2).

Serial No	CONJUC.	ISA
1	0	0
32	0	16
2	1	1
14	1	17
15	2	2
29	2	18
3	3	3
10	3	19
27	4	20
30	4	4
16	5	5
18	5	21
11	6	6
24	6	22
4	7	7
20	7	23
13	8	24
31	8	8
9	9	25
28	9	9
17	10	10
26	10	26
19	11	11
23	11	27
8	12	28
12	12	12
22	13	29
25	13	13
7	14	30
21	14	14
5	15	15
6	15	31

Fig. 5.2. The De Bruijn Array is sorted by the conjugate vector.

The distance between the conjugate element is calculated by subtracting the two serial numbers of the conjugate elements creating in this way the third vector with the distances.

Distance	Serial No	CONJUC.	ISA
31	1	0	0
31	32	0	16
12	2	1	1
12	14	1	17
14	15	2	2
14	29	2	18
7	3	3	3
7	10	3	19
3	27	4	20
3	30	4	4
2	16	5	5
2	18	5	21
13	11	6	6
13	24	6	22
16	4	7	7
16	20	7	23
18	13	8	24
18	31	8	8
19	9	9	25
19	28	9	9
9	17	10	10
9	26	10	26
4	19	11	11
4	23	11	27
4	8	12	28
4	12	12	12
3	22	13	29
3	25	13	13
14	7	14	30
14	21	14	14
1	5	15	15
1	6	15	31

Fig. 5.3. The Array with the distances between the conjugates.

- The matrix is finally restored to its original shape sorting it once more this time by serial number (Fig. 5.3). It must be noted that, before the sorting, it is important to replace the Distance matrix from the calculated formula values with fixed ones.

Distance	Serial No	CONJUC.	ISA
31	1	0	0
12	2	1	1
7	3	3	3
16	4	7	7
1	5	15	15
1	6	15	31
14	7	14	30
4	8	12	28
19	9	9	25
7	10	3	19
13	11	6	6
4	12	12	12
18	13	8	24
12	14	1	17
14	15	2	2
2	16	5	5
9	17	10	10
2	18	5	21
4	19	11	11
16	20	7	23
14	21	14	14
3	22	13	29
4	23	11	27
13	24	6	22
3	25	13	13
9	26	10	26
3	27	4	20
19	28	9	9
14	29	2	18
3	30	4	4
18	31	8	8
31	32	0	16

Fig.5.3. The Matrix is restored in the original De Bruijn sequences accompanied by the vectors required for the cross join pair operation.

Once this is done, the pairs are selected to satisfy proposition 3 e.g. to provide maximum diameter d. To achieve this, the selection must be done between the elements of the ISA with the shorter distance between them which are interleaved.

The two conjugate pairs are marked and a cross join pair operation is performed creating a new ISA with De Bruijn sequences properties Fig. 5.4).

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

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

Fig. 5.4. Selection of the pairs and cross join par operation.

The two sequences primary and derived are compared by XORing their corresponding elements of the arrays. The product of the comparison is shown in fig.5.5 together with the calculation of the Longer Common Subsequence between them.

DISTANCE	CONJUC.	ISA	DBS		NISA	NDBS	Verif.	LCS=	30
31	0	0	0		0	0	0		
12	1	1	0		1	0	0		
7	3	3	0		3	0	0		
16	7	7	0		7	0	0		
1	15	15	0		15	0	0		
1	15	31	1		31	1	0		
14	14	30	1		30	1	0		
5	12	28	1		28	1	0		
20	9	25	1		25	1	0		
7	3	19	1		19	1	0		
13	6	6	0		6	0	0		
5	12	12	0		12	0	0		
18	8	24	1		24	1	0		
12	1	17	1		17	1	0		
14	2	2	0		2	0	0		
2	5	5	0		5	0	0		
9	10	10	0		10	0	0		
2	5	21	1		21	1	0		
4	11	11	0		11	0	0		
16	7	23	1		22	1	0		
14	14	14	0		13	0	0		
3	13	29	1		27	1	0		
4	11	27	1		23	1	0		
13	6	22	1		14	0	-1		
3	13	13	0		29	1	1		
9	10	26	1		26	1	0		
3	4	20	1		20	1	0		
20	9	9	0		9	0	0		
14	2	18	1		18	1	0		
3	4	4	0		4	0	0		
18	8	8	0		8	0	0		
31	0	16	1		16	1	0		
						Absolute Δ	**2**		

Fig. 5.5 Final table with the two Be Bruijn Sequences and their LCS.

With the implementation of the described algorithm in chapter 6, it is observed that compliance is achievable in many cases of sequences with maximum sharing e.g. 2^n-2 in most samples of n=5, and to a lesser degree with higher n's with minimum 2^n -6.

5.2 The longest length of shared subsequences

The process for the calculation of the longest common subsequence of the two De Bruijn sequences is done with the comparison of their binary form.

Since the preparatory work for practical reasons is done using the ISA form it is required to it to binary.

This is achieved by comparing the elements of the ISA to 2^{n-1}. All the elements which are lower than the 2^{n-1} are replaced with 0 otherwise is 1 or in the particular example using the statement =IF(K10>=16,1,0) where k10 is the ISA element, 16 the 25-1, 1 if is TRUE,) and 0 if is False.

This is nicely presented in Fig. 5.5. where the vectors:

ISA and DBS represent the primary sequence,

NISA, and NDBS represents the derived sequence,

Verif., represents the product of the comparison between the two sequences (=ISA-NISA),

The absolute Δ (=SUMSQ(L9:L40)) , represents the absolute value of the Verif. vector and the LCS the Longest Common subsequence (=32 -L41).

Chapter 6

Implementation of the algorithm - evaluation of its effectiveness.

6.1 Derivation of the corresponding subsequence with Longer shared Common Subsequence (LCS),

The process of analyzing De Bruijn sequences is performed in the following steps:

Random construction of the primary De Bruin Sequences using an algorithm supported by a c program. The construction will investigate sequences of order n=5, n=6, n=7, n=8, and n=9.

The primary sequences will be used a the basis to derive the corresponding subsequence that shares Longer Common Subsequence (LCS). The objective is to use the cross pair operations to produce optimal LCS that ideally has a 2^n-2 correspondence or the closest to it. During the process, the maximum diameter (d) concept will be tested to prove that using the maximum diameter (d) could produce optimal LCS.

The primary and the derived sequence will be directly tested for Maximum LCS, and through the M&S tool to calculate the Linear Complexity.

Also, the primary and the derived sequences shall be conditioned in such a way to be processed via the SAGEmath tool to calculate the cryptographic properties of those Boolean functions that "produce" De Bruijn sequences. These properties are the :

- algebraic_degree,
- nonlinearity,

- algebraic_immunity

Furthermore, the cryptographic properties of the primary and the derived sequences will be compared among themselves and to their max degree achieved and establish conditions that could assist the production of strong cryptographic generators by knowing in advance their properties, the improvement or risks of the corresponding sequences that are sharing long parts of their original patterns.

Finally, the results will be formulated and tabulated in a friendly way to be used by the designers of NLFSR's so that could rate their cryptographic criteria and adjust them accordingly by considering their closest longer sharing subsequence.

6.2 Random construction of the primary De Bruin Sequences,

There are several methods for the construction of primary De Bruijn sequences. These were described in chapter 3 and comprises the construction of binary De Bruijn sequences using:

- De Bruijn Graphs
- Necklace (prefer-one)
- Arithmetic balances
- Suffix Arrays.

The latter is the method used by the c based algorithm for the construction of the primary De Bruijn sequences which will be used in the analysis.

The program in C does the following when you run it: it "randomly" generates a suffix array corresponding to a De Bruijn sequence. By "random" I mean that each time it produces another suffix array because it executes a probabilistic and not deterministic algorithm.

On the screen, it prints the Suffix Array (SA) and the Inverse Suffix Array (ISA) while creating a text file that writes the corresponding De Bruijn Sequence (DBS) and ISA .

The size of DBS is specified in the code - cf. at the beginning the commands

#define N is the size of the sequence

#define n is the degree of the sequence.

The program runs to produce:

- 20 DBS's of n=5,
- 10 DBS's of n=6,
- 10 DBS's of n=7,
- 10 DBS's of n=8,
- 10 DBS's of n=9.

The above sample is deemed adequate to provide sound results and facilitate the extraction of conclusions about the cryptographic characteristics of the sequences under investigation and the corresponding sequences that are sharing a common part of an optimal length.

```
00000111110011000101011101101001
0 1 3 7 15 31 30 28 25 19 6 12 24 17 2 5 10 21 11 23 14 29 27 22 13 26 20 9 18 4 8 16
```

Fig.6.1 Sample of an n=5 algorithm2.c output. (De Bruijn sequence and the Inverse Suffix Array)

To appreciate the size and complexity of the DBS and ISA as n increase to 6,7,8 and 9 the output of algorithm2.c is appended here below:

0000001010100001101111110001111010010001001100101110011101101011
0 1 2 5 10 21 42 20 40 16 33 3 6 13 27 55 47 31 63 62 60 56 49 35 7 15 30 61 58 52 41 18 36 8 17
34 4 9 19 38 12 25 50 37 11 23 46 28 57 51 39 14 29 59 54 45 26 53 43 22 44 24 48 32

Fig.6.2 Sample of an n=6 algorithm2.c output. (De Bruijn sequence and the Inverse Suffix Array)

0000000101101000001100010001111010100100111110011100101000101010111011110001 1
0111000011101001100100001001011111110110110011010 11

0 1 2 5 11 22 45 90 52 104 80 32 65 3 6 12 24 49 98 68 8 17 35 71 15 30 61 122 117 106 84 41 82
36 73 19 39 79 31 62 124 121 115 103 78 28 57 114 101 74 20 40 81 34 69 10 21 42 85 43 87 46
93 59 119 111 94 60 120 113 99 70 13 27 55 110 92 56 112 97 67 7 14 29 58 116 105 83 38 76 25
50 100 72 16 33 66 4 9 18 37 75 23 47 95 63 127 126 125 123 118 109 91 54 108 89 51 102 77 26
53 107 86 44 88 48 96 64

Fig.6.3 Sample of an n=7 algorithm2.c output. (De Bruijn sequence and the Inverse Suffix Array)

0000000010100011010010000001110100111000111101000101100000100111100101001100100 1
0101101000001100011001110110110101010010111100001111111100010101011101111001111
1011101010000110110011000010111001101111011000100001000111001000100100110101111 1
101011001011011 1

0 1 2 5 10 20 40 81 163 70 141 26 52 105 210 164 72 144 32 64 129 3 7 14 29 58 116 233 211 167 78
156 56 113 227 199 143 30 61 122 244 232 209 162 69 139 22 44 88 176 96 193 130 4 9 19 39 79
158 60 121 242 229 202 148 41 83 166 76 153 50 100 201 146 37 74 149 43 86 173 90 180 104 208
160 65 131 6 12 24 49 99 198 140 25 51 103 206 157 59 118 237 219 182 109 218 181 106 213 170
84 169 82 165 75 151 47 94 188 120 240 225 195 135 15 31 63 127 255 254 252 248 241 226 197
138 21 42 85 171 87 174 93 187 119 239 223 190 124 249 243 231 207 159 62 125 251 247 238 221
186 117 234 212 168 80 161 67 134 13 27 54 108 217 179 102 204 152 48 97 194 133 11 23 46 92
185 115 230 205 155 55 111 222 189 123 246 236 216 177 98 196 136 16 33 66 132 8 17 35 71 142
28 57 114 228 200 145 34 68 137 18 36 73 147 38 77 154 53 107 215 175 95 191 126 253 250 245
235 214 172 89 178 101 203 150 45 91 183 110 220 184 112 224 192 128

Fig.6.4 Sample of an n=8 algorithm2.c output. (De Bruijn sequence and the Inverse Suffix Array)

00000000010010000110000100110000010110100101100111110110101000100010100011100100
10101001000100101110010111111010111011001000001101101111110111110010101100101000
00101110101011010110000111000010101010111000110011100110110001010111101010011100
01001110110110100011000100000000110010011010111110100110011000111110000011101011 0
11101110011101001001001111000011010011111111100111101110100000100001000110111000
00011110100010111100100011110011001011011001101000011111100010110001101010100000
0101001010011011110001110111101 1

0 1 2 4 9 18 36 72 144 289 67 134 268 24 48 97 194 388 265 19 38 76 152 304 96 193 386 261 11 22
45 90 180 361 210 421 331 150 300 89 179 359 207 415 318 125 251 502 493 474 437 362 212 424
337 162 324 136 273 34 69 138 276 40 81 163 327 142 284 57 114 228 457 402 293 74 149 298 84
169 338 164 328 145 290 68 137 274 37 75 151 302 92 185 370 229 459 407 303 95 191 383 254 509
506 501 491 471 430 349 187 374 236 473 434 356 200 400 288 65 131 262 13 27 54 109 219 439
367 223 447 382 253 507 503 495 479 446 380 249 498 485 458 405 299 86 172 345 178 357 202
404 296 80 161 322 133 267 23 46 93 186 373 234 469 427 342 173 346 181 363 214 428 344 176
353 195 391 270 28 56 112 225 450 389 266 21 42 85 170 341 171 343 174 348 184 369 227 454 396
281 51 103 206 412 313 115 230 461 411 310 108 216 433 354 197 394 277 43 87 175 350 189 378
245 490 468 425 339 167 334 156 312 113 226 452 393 275 39 78 157 315 118 237 475 438 365 218
436 360 209 419 326 140 280 49 98 196 392 272 32 64 128 257 3 6 12 25 50 100 201 403 294 77 154
309 107 215 431 351 190 381 250 500 489 467 422 332 153 307 102 204 408 305 99 199 399 287 62
124 248 496 480 449 387 263 14 29 58 117 235 470 429 347 183 366 221 443 375 238 476 441 371
231 462 413 314 116 233 466 420 329 146 292 73 147 295 79 158 316 120 240 481 451 390 269 26
52 105 211 423 335 159 319 127 255 511 510 508 505 499 487 463 414 317 123 247 494 477 442
372 232 464 416 321 130 260 8 16 33 66 132 264 17 35 70 141 283 55 110 220 440 368 224 448 385
259 7 15 30 61 122 244 488 465 418 325 139 279 47 94 188 377 242 484 456 401 291 71 143 286 60
121 243 486 460 409 306 101 203 406 301 91 182 364 217 435 358 205 410 308 104 208 417 323
135 271 31 63 126 252 504 497 482 453 395 278 44 88 177 355 198 397 282 53 106 213 426 340 168
336 160 320 129 258 5 10 20 41 82 165 330 148 297 83 166 333 155 311 111 222 444 376 241 483
455 398 285 59 119 239 478 445 379 246 492 472 432 352 192 384 256

Fig.6.5 Sample of an n=9 algorithm2.c output. (De Bruijn sequence and the Inverse Suffix Array)

All the primary sequences are presented in Appendix A1- A5.

6.3 Derivation of the corresponding subsequence with Longer shared Common Subsequence (LCS),

The derivation of the optimal Longest Common Subsequence (LCS) will be done with the aid of a visual record methodology using the Microsoft XL spreadsheet and is divide into three stages:

- Preparation of the field table.
- Performing the cross-join pair operation.

- Verifying the degree of the Longest Common Subsequence (LCS) between the initial and the derived sequence.

The same could be done using programming tools however due to lack of fluency in programming it is opted to use the method of visual records.

This is a simple and comprehensible way for the reader. The manual operation however restricts the size of the investigation.

The examination starts with the placement of the binary and the ISA form o the DBS on the XL sheet. The binary form is recorded only for reference since the ISA could provide the info required to expand the elements of the sequence to perform the investigation. Additionally, working with the binary form of the BDS is very complicated and confusing since the ordinary mind is more familiar to work with the decimal rather than binary. Preparation of the field table,

To perform the investigation the following fields are required:

A/A	DISTANCE	CONJUC.	ISA	DBS
1	31	0	0	0
2	12	1	1	0
3	7	3	3	0
4	16	7	7	0
5	1	15	15	0
6	1	15	31	1
7	14	14	30	1
8	5	12	28	1
9	20	9	25	1
10	7	3	19	1
11	13	6	6	0
12	5	12	12	0
13	18	8	24	1

Fig. 6.6. Table with the fields required for performing the cross-join pair ops.

ISA : The Inverse Suffix Array of the sequence under investigation.

CONJUC: The pairs of the conjugates of the DBS.

DISTANCE: The distance between the pair of the two conjugates.

A/A: The reference number that is used to facilitate the calculation of the DISTANCE.

DBS: The De Bruijn sequence under examination.

ISA is the base parameters, and the rest are derived as follows:

Conjugates are produced by calculating the mod(ISA, 2^{n-1}). In the case of n=5 is the mod(ISA, 16),

and in the case of Fig. X.6 is the mod(0,16)= 0, mod(7,16)=7, mod(31,16)= 15, mod(25,16)=9.

In this way, the table is split into two lists identifying the list of the two conjugates.

Distance is calculated by sorting the total table by the CONJUC column and distance between the two

adjacent rows.

The table is restored to its initial form by sorting it by A/A column.

DBS is formed by applying the command =IF(ISA<=15,0,1) where ISA is the corresponding cell of the

ISA column examined whether it belongs to the first half of the sequence which is presented with "0" or

the second half which is presented with "1".

6.2 Performing the cross-pair operation,

The cross-pair operation is performed after the selection of the appropriate conjugated pairs that are satisfying the following conditions:

- The pairs must be interleaved/crossed.
- Maximize the diameter d.

In doing so, the table is scanned and the pairs that are crossing each other with the shortest DISTANCEs are selected. Assuming the two DISTANCEs are l and k then l+k is min.

Also, the process of selection must account for the way that the two pairs are crossing, e.g. l+k is min closer to the biggest of the two parameters plus one.

After the selection of the pairs, the crossing operation is performed by:

Painting the two terms of the two pairs with different colors e.g., yellow for pair one and green for pair two.

A/A	DISTANCE	CONJUC.	ISA	DBS
1	31	0	0	0
2	12	1	1	0
3	7	3	3	0
4	16	7	7	0
5	1	15	15	0
6	1	15	31	1
7	14	14	30	1
8	5	12	28	1
9	20	9	25	1
10	7	3	19	1
11	13	6	6	0
12	5	12	12	0
13	18	8	24	1
14	12	1	17	1
15	14	2	2	0
16	2	5	5	0
17	9	10	10	0
18	2	5	21	1
19	4	11	11	0
20	16	7	23	1
21	14	14	14	0
22	3	13	29	1
23	4	11	27	1
24	13	6	22	1
25	3	13	13	0
26	9	10	26	1
27	3	4	20	1
28	20	9	9	0
29	14	2	18	1
30	3	4	4	0
31	18	8	8	0
32	31	0	16	1

Summary | n=5 | n=6 | n=7 | n=8 | n

Fig. 6.7: Selection of the conjugate pairs.

Copying the first section of the sequence until and including the first term of the conjugate (first term of the DBS until the first yellow term) and paste it to the column where the derived DBS will be formed.

ISA	DBS			NISA	NDBS
0	0			0	0
1	0			1	0
3	0			3	0
7	0			7	0
15	0			15	0
31	1			31	1
30	1			30	1
28	1			28	1
25	1			25	1
19	1			19	1
6	0			6	0
12	0			12	0
24	1			24	1
17	1			17	1
2	0			2	0
5	0			5	0
10	0			10	0
21	1			21	1
11	0			11	0

Fig.6.8: Table showing the first action of the cross-join pair operation.

Copying the terms following the end of the first pair (starting after the second yellow term) going to the second term of the second pair (until the green painted term) and paste it as the continuation of the ii.

22	1
13	0

11	0
22	1
13	0

Fig.6.9: Table showing the second action of the cross-join pair operation.

Copying the terms following the first term of the first pair (terms following the first painted yellow term) and paste it as a continuation of iii.

27	1

13	0
27	1

Fig.610: Table showing the third act of the cross-join pair operation.

Copying the terms following the first term of the second pair (painted green) and paste them as a continuation of iv.

22	1
13	0

27	1
23	1
14	0
29	1

Fig.6.11: Table showing the fourth action of the cross-join pair operation.

Finally, copying the terms following the second term of the second pair (painted green) until the end of the sequence and paste it as a continuation of

26	1
20	1
9	0
18	1
4	0
8	0
16	1

26	1
20	1
9	0
18	1
4	0
8	0
16	1

Fig.612: Table showing the Final action of the cross-join pair operation.

The final picture of the operations is:

A/A	DISTANCE	CONJUC.	ISA	DBS	NISA	NDBS	Ver.	LCS	30
1	31	0	0	0	0	0	0		
2	12	1	1	0	1	0	0		
3	7	3	3	0	3	0	0		
4	16	7	7	0	7	0	0		
5	1	15	15	0	15	0	0		
6	1	15	31	1	31	1	0		
7	14	14	30	1	30	1	0		
8	5	12	28	1	28	1	0		
9	20	9	25	1	25	1	0		
10	7	3	19	1	19	1	0		
11	13	6	6	0	6	0	0		
12	5	12	12	0	12	0	0		
13	18	8	24	1	24	1	0		
14	12	1	17	1	17	1	0		
15	14	2	2	0	2	0	0		
16	2	5	5	0	5	0	0		
17	9	10	10	0	10	0	0		
18	2	5	21	1	21	1	0		
19	4	11	11	0	11	0	0		
20	16	7	23	1	22	1	0		
21	14	14	14	0	13	0	0		
22	3	13	29	1	27	1	0		
23	4	11	27	1	23	1	0		
24	13	6	22	1	14	0	-1		
25	3	13	13	0	29	1	1		
26	9	10	26	1	26	1	0		
27	3	4	20	1	20	1	0		
28	20	9	9	0	9	0	0		
29	14	2	18	1	18	1	0		
30	3	4	4	0	4	0	0		
31	18	8	8	0	8	0	0		
32	31	0	16	1	16	1	0		
							2		

Fig. 6.13: Final picture after the completion of the cross-join pair operation.

During the cross-pair operation, the correctness of the process is tested by testing the size of the cross-pair operation in the initial DBS is the same as the derived one (NDBS).

All the results are presented in Appendix B1-B5 and the attached XL sheets.

6.4 Verifying the degree of the Longest Common Subsequence (LCS) between the initial and the derived sequence,

The degree of the sharing between the initial and the derived sequence is calculated by comparing one to one of the corresponding columns marked in column ver.

The ver column is the comparison between the initial DBS and NDBS that produces results 0 indicating equality or 1 and -1 indicating the difference. The sum of the absolute values of the ver column ***sumq(k9:k40)*** indicates the degree of the resemblance between the two sequences, original and derived.

The degree of resemblance **LCS** is ***2^n-sumq(k9:k40)*** *as presented in Fig.X.13*

Following the above methodology, the algorithm for deriving De Bruijn sequences from primary with objective the longest common subsequences is implemented to a sample of:

- 20 sequences of n=5,
- 10 sequences of n=6,
- 10 sequences of n=7,
- 10 sequences of n=5,
- 10 sequences of n=5.

The total operation is filed in the Excel sheet in Appendix B1-B5 which is submitted with this thesis and constitutes an integral part of the study.

Chapter 7

Cryptographic properties of relevant De Bruijn generators,

7. Testing:

In this chapter we describe the methodology we adopted, towards evaluating for each pair of De Bruin sequence which share the (possibly) longest common subsequence, their linear complexities (i.e. to find out whether such almost identical De Bruijn sequences have large discrepancies in their linear complexities), as well as the behavior of the corresponding Boolean functions that generated them, in terms of the following cryptographic criteria: :

a. algebraic_degree,
b. nonlinearity,
c. algebraic_immunity

7.1. Linear Complexity,

The **linear complexity** of a **de Bruijn sequence** is the degree of the shortest **linear** recursion which generates the **sequence**.
It can be calculated using the Berlekamp-Massey algorithm of Cryptool 2.
This is an algorithm that will find the shortest linear feedback shift register (LFSR) for a given binary output sequence. The algorithm will also find the minimal polynomial of a linearly recurrent sequence in an arbitrary field.
The tool accepts as input the DBS and outputs the C= shortest LFSR and the minimal polynomial of a linearly recurrent sequence in an arbitrary file e.g. Linear Complexity as defined above and L= Minimal Lenght.

The process comprises the setup of four modules, one Berlekamp-Massey engine with input the DBS sequence and output Minimal Lenght L and the Feedback Polynomial C (Fig. 7.1)

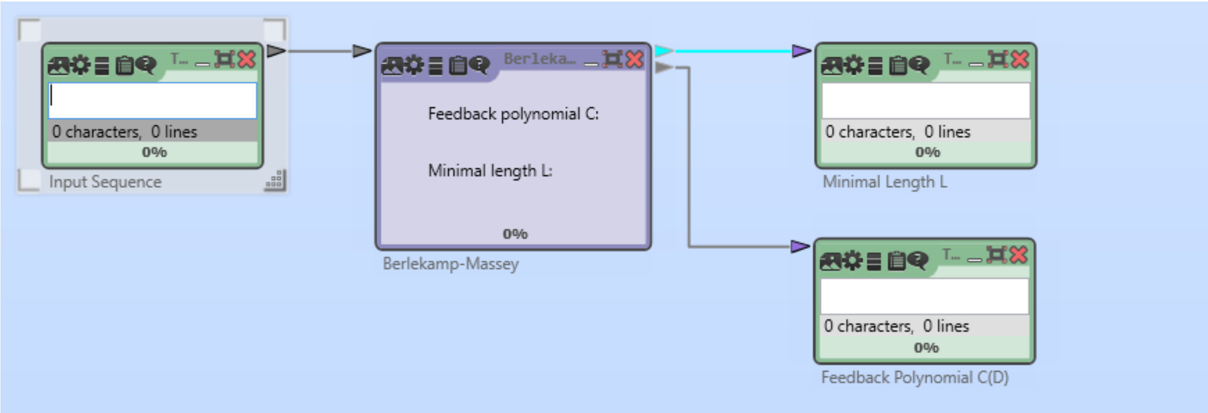


Fig. 7.1: The four modules Berlekamp-Massey initial setup.

The input is the DBS under examination. However, because DBS's are periodic, to get the correct value of Linear complexity you have to enter the sequence twice. E.g.

Considering the example of Fig. X.13 the DBS is

00000111110011000101011101101001,

then the input to B-M engine must be:
0000011111001100010101110110100100000111110011000101011101101001

With the above as input, the output of the B-M tool are:

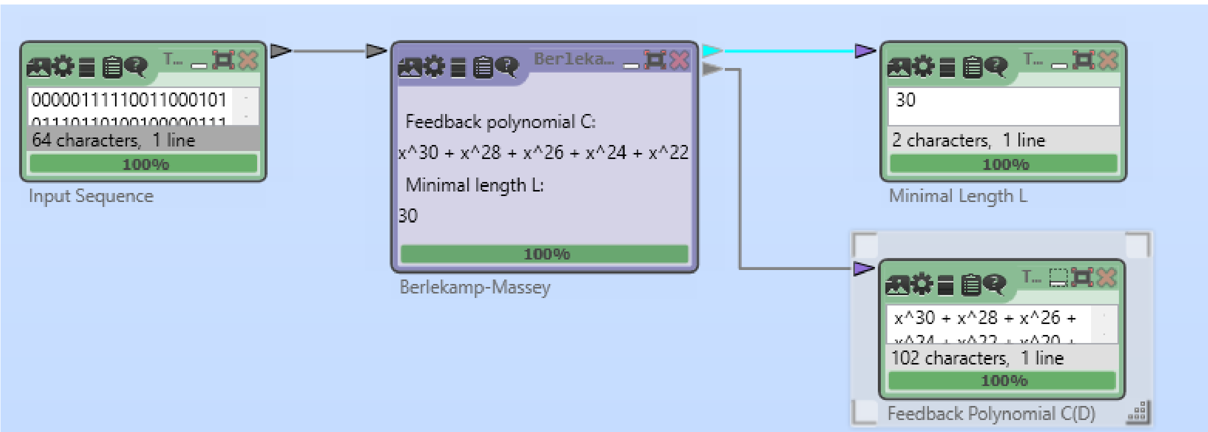


Fig.7.2: The four modules Berlekamp-Massey evaluating an n=5 De Bruijn sequence.

L=30,

C= $x^{30} + x^{28} + x^{26} + x^{24} + x^{22} + x^{20} + x^{18} + x^{16} + x^{14} + x^{12} + x^{10} + x^8 + x^6 + x^4 + x^2 + 1$.

Following the above methodology, the Linear complexity of the DBS (initial and Derived) is calculated and a sample tabulated below:

	De Bruijn Sequence n=5	Linear Complexity
original 1	00000111110011000101011101101001	30
derived 1	00000111110011000101011011101001	31
original 2	00000101011111000111011001001101	31
derived 2	00000101011111000111011001101001	27
original 3	00000110001010011101011011111001	24
derived 3	00000110001010011101101011111001	31
original 4	00000100011001011010100111011111	31
derived 4	00000100011001010110100111011111	30
original 5	00000111001000101111101010011011	28
derived 5	00000111001000101011111010011011	24

Fig, 7.3: Table with a sample of the Linear Complexity for primary and derived De Bruijn sequences.

All the results are presented in Appendix C1-C5.Examination of the Boolean Function that produces the DBS's

7.2 Examination of the Boolean Function that produces the DBS's:

In the examination of the other cryptographic criteria, e.g. algebraic_degree, nonlinearity, algebraic_immunity, the process is not so direct as the linear complexity because is not done on the actual DBS but with their corresponding Boolean Function.

For this reason, the DBS's must be converted to their corresponding Boolean Function.

The process for constructing the Boolean Function that produces a DBS is a follows:

Use the ISA form of the DBS and convert it into Decimal using the command DEC2Bin(ISA,5).

Add the Next BiT (NXBIT) Column, by shifting the DBS n rows upward and complete the last n bits with zero's,

ISA	DBS	NXTBIT	
0	0	1	00000
1	0	1	00001
3	0	1	00011
7	0	1	00111
15	0	1	01111
31	1	0	11111
30	1	0	11110
28	1	1	11100
25	1	1	11001
19	1	0	10011
6	0	0	00110
12	0	0	01100

Fig. 7.4: Table showing the conversion of a De Bruijn sequence from ISA to Binary.

Then using the Text To Column facility of the Data in the Microsoft Exel binary terms as ISA are expanded in the corresponding columns.

ISA	DBS	NXTBIT		X0	X1	X2	X3	X4
0	0	1	00000	0	0	0	0	0
1	0	1	00001	0	0	0	0	1
3	0	1	00011	0	0	0	1	1
7	0	1	00111	0	0	1	1	1
15	0	1	01111	0	1	1	1	1
31	1	0	11111	1	1	1	1	1
30	1	0	11110	1	1	1	1	0
28	1	1	11100	1	1	1	0	0
25	1	1	11001	1	1	0	0	1
19	1	0	10011	1	0	0	1	1
6	0	0	00110	0	0	1	1	0
12	0	0	01100	0	1	1	0	0

Fig. 7.5: Table showing the expansion of a binary De Bruin sequence into columns to facilitate sorting.

Sort the table of fig.x.18 backward starting from x4->x0 using the XL command =SORTBY(S9:AA40,AA9:AA40,1,Z9:Z40,1,Y9:Y40,1,X9:X40,1,W9:W40,1).

The execution of the command will reformat the Next Bit of DBS expansion to Boolean Function as shown in Fig 7.6 in yellow.

ISA	DBS	NXTBIT		X0	X1	X2	X3	X4									
0	0	1	00000	0	0	0	0	0	0	0	**1**	00000	0	0	0	0	0
1	0	1	00001	0	0	0	0	1	16	1	**0**	10000	1	0	0	0	0
3	0	1	00011	0	0	0	1	1	8	0	**0**	01000	0	1	0	0	0
7	0	1	00111	0	0	1	1	1	24	1	**1**	11000	1	1	0	0	0
15	0	1	01111	0	1	1	1	1	4	0	**0**	00100	0	0	1	0	0
31	1	0	11111	1	1	1	1	1	20	1	**1**	10100	1	0	1	0	0
30	1	0	11110	1	1	1	1	0	12	0	**0**	01100	0	1	1	0	0
28	1	1	11100	1	1	1	0	0	28	1	**1**	11100	1	1	1	0	0
25	1	1	11001	1	1	0	0	1	2	0	**1**	00010	0	0	0	1	0
19	1	0	10011	1	0	0	1	1	18	1	**0**	10010	1	0	0	1	0
6	0	0	00110	0	0	1	1	0	10	0	**1**	01010	0	1	0	1	0
12	0	0	01100	0	1	1	0	0	26	1	**0**	11010	1	1	0	1	0

Fig. 7.6: Table showing the sorting of the binary De Bruijn sequences converted into Boolean Function format.

All the results are presented in Appendix C1-C5 including the corresponding XL sheets.

The process will be repeated for all n=5 -> n=9 for the initial and derived DBS's and their horizontal presentation separated by commas will be processed by the SAGEmath tool calculating the cryptographic parameters of the Boolean Functions that are producing The DBS's as below:

Sage: from sage.crypto.boolean_function import BooleanFunction

sage: f = BooleanFunction

([1,0,0,1,0,1,0,1,1,0,1,0,0,1,1,0,1,0,0,1,0,1,0,1,1,0,1,0,1,0,1,0])

(is the output of the truth table)

sage: f.algebraic_degree ()

(will show us the algebraic degree of the function - ideally, the maximum value it can get is n-1).

sage: f.nonlinearity ()

(will show us the nonlinearity of the function - ideally, the maximum value it can get is 2n-1 - 2 (n / 2) -1 when n is even, and about 2n-1 - 2 (n-1/2) -1 when n is odd).

sage: f.algebraic_immunity ()

(will show us the algebraic robustness of the function - ideally, the maximum value it can get is n / 2 when n is even, and (n-1) / 2 when n is odd).

```
from sage.crypto.boolean_function import BooleanFunction
f=BooleanFunction([1,0,0,1,0,1,0,1,1,0,1,0,0,1,1,0,1,0,0,1,0,1,0,1,1,0,1,0,1,0,1,0])
f.algebraic_degree()
f.nonlinearity()
f.algebraic_immunity()

4
6
2
```

Fig. 7.7; Sample image from the tool that evaluates the crypto properties of the Boolean Function the produces De Bruijn sequences.

A sample table is presented in fig.7.8.

True table- De Bruijn Sequence n=5	LCS	Algebraic Degree	Non Linearity	Algebraic Immunity	Δ Non linearity	Δ Algebraic Immunity
1,0	30	4	6	2	0	0
1,0,0,1,0,1,0,1,1,0,1,0,0,1,1,0,1,0,0,1,0,1,1,0,1,0,0,1,1,0,1,0		4	6	2		
1,0,0,1,1,0,1,0,1,0,1,0,1,0,1,0,0,1,1,0,0,1,0,1,1,0,1,0,0,1,1,0	30	4	10	3	4	1
1,0,0,1,0,1,1,0,1,0,1,0,1,0,1,0,0,1,0,1,0,1,0,1,1,0,1,0,0,1,1,0		4	6	2		
1,0,0,1,0,1,0,1,1,0,0,1,0,1,1,0,1,0,1,0,0,1,1,0,0,1,0,1,0,1,1,0	30	4	6	2	4	1
1,0,0,1,0,1,0,1,1,0,0,1,0,1,1,0,1,0,1,0,0,1,0,1,0,1,1,0,0,1,1,0		4	10	3		
1,0,1,0,0,1,1,0,0,1,0,1,0,1,1,0,0,1,1,0,1,0,0,1,0,1,0,1,0,1,1,0	30	4	10	3	4	1
1,0,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,1,0,1,0,1,0,1,0,1,1,0		4	6	2		

Fig.7.8: Sample of a summary table with the crypto properties of the primary and derived De Bruijn sequences.

All the results are presented in Appendix A.e including the corresponding pdf files.

Additionally, the Excell sheets with the corresponding calculations are also submitted with this Thesis and constitute an integral part of this study.

Chapter 8

Results and Conclusions

Results:

Following the analysis presented in chapter 7, the results of the operations are summarized here below to draw conclusions based on the theory behind and the experimental outcome.
To this effect, the outcome of the analysis done on the sequences of degree n=5,6,7,8,9 are tabulated in pairs to stipulate the:

- **LCS,** (Longer Common Subsequence) between the primary and the derived sequence.
- **Linear Complexity** of the two sequences and their delta (Δ difference),
- **Algebraic Degree,** corresponding pair,
- **Non-Linearity** of the pair and their delta (Δ difference),
- **Algebraic Immunity** of the pair and their delta ((Δ difference).

n=5	Linear Complexity	LCS	Algebraic Degree	Non Linearity	Algebraic Immunity	Δ Linear Complexity	Δ Non linearity	Δ Algebraic Immunity
original 1	30	30	4	6	2	1	0	0
derived 1	31		4	6	2			
original 2	31	30	4	10	3	4	4	1
derived 2	27		4	6	2			
original 3	24	30	4	6	2	7	4	1
derived 3	31		4	10	3			
original 4	31	30	4	10	3	1	4	1
derived 4	30		4	6	2			
original 5	28	30	4	10	3	4	4	1
derived 5	24		4	6	2			
original 6	30	30	4	6	2	1	4	1
derived 6	31		4	10	3			
original 7	31	28	4	6	2	0	4	1
derived 7	31		4	10	3			
original 8	32	30	4	6	2	1	0	0
derived 8	31		4	6	2			
original 9	31	28	4	10	3	3	4	1
derived 9	28		4	6	2			
original 10	30	30	4	6	2	7	0	0
derived 10	23		4	6	2			
original 11	30	30	4	6	2	2	4	1
derived 11	28		4	10	3			
original 12	30	26	4	6	2	1	0	0
derived 12	29		4	6	2			
original 13	31	30	4	10	3	0	0	0
derived 13	31		4	10	3			
original 14	31	30	4	6	2	7	0	0
derived 14	24		4	6	2			
original 15	28	30	4	6	2	2	0	0
derived 15	30		4	6	2			
original 16	31	30	4	6	2	0	0	0
derived 16	31		4	6	2			
original 17	30	30	4	6	2	1	4	1
derived 17	31		4	10	3			
original 18	30	30	4	10	3	2	4	1
derived 18	28		4	6	2			
original 19	31	26	4	6	2	0	4	1
derived 19	31		4	10	3			
original 20	29	30	4	10	3	1	0	0
derived 20	30		4	10	3			

Fig.8.1. Summary of experimental results for n=5.

n=6	Linear Complexity	LCS	Algebraic Degree	Non Linearity	Algebraic Immunity	Δ Linear Complexity	Δ Non linearity	Δ Algebraic Immunity
original 1	61	60	5	18	3	2	0	0
derived 1	63		5	18	3			
original 2	60	60	5	22	3	3	4	0
derived 2	63		5	18	3			
original 3	59	58	5	22	3	3	4	0
derived 3	62		5	18	3			
original 4	63	58	5	18	3	1	4	0
derived 4	62		5	22	3			
original 5	62	60	5	14	3	1	0	0
derived 5	61		5	14	3			
original 6	63	58	5	22	3	3	0	0
derived 6	60		5	22	3			
original 7	62	60	5	18	3	1	4	0
derived 7	63		5	22	3			
original 8	61	60	5	22	3	2	4	0
derived 8	63		5	18	3			
original 9	63	58	5	14	3	0	4	0
derived 9	63		5	18	3			
original 10	63	60	5	14	3	1	4	0
derived 10	62		5	18	3			

Fig.8.2. Summary of experimental results for n=6.

n=7	Linear Complexity	LCS	Algebraic Degree	Non Linearity	Algebraic Immunity	Δ Linear Complexity	Δ Non linearity	Δ Algebraic Immunity
original 1	124	124	6	42	3	1	4	0
derived 1	125		6	46	3			
original 2	126	122	6	46	3	1	0	0
derived 2	127		6	46	3			
original 3	122	126	6	42	3	4	0	0
derived 3	126		6	42	3			
original 4	127	126	6	46	3	1	4	0
derived 4	126		6	42	3			
original 5	126	120	6	46	3	1	0	0
derived 5	125		6	46	3			
original 6	123	126	6	46	3	3	0	0
derived 6	126		6	46	3			
original 7	127	126	6	46	3	1	0	0
derived 7	126		6	46	3			
original 8	126	124	6	46	3	0	4	0
derived 8	126		6	50	3			
original 9	126	122	6	46	3	1	4	0
derived 9	127		6	42	3			
original 10	126	124	6	42	3	1	4	0
derived 10	125		6	38	3			

Fig.8.3. Summary of experimental results for n=7.

n=8	Linear Complexity	LCS	Algebraic Degree	Non Linearity	Algebraic Immunity	Δ Linear Complexity	Δ Non linearity	Δ Algebraic Immunity
original 1	253	250	7	94	4	1	4	0
derived 1	254		7	90	4			
original 2	252	242	7	98	4	2	2	0
derived 2	254		7	96	4			
original 3	255	252	7	98	4	4	4	0
derived 3	251		7	102	4			
original 4	254	250	7	102	4	0	4	0
derived 4	254		7	98	4			
original 5	255	252	7	102	4	0	4	0
derived 5	255		7	98	4			
original 6	251	248	7	98	4	1	0	0
derived 6	252		7	98	4			
original 7	254	252	7	98	4	1	4	0
derived 7	255		7	102	4			
original 8	255	252	7	94	4	1	0	0
derived 8	254		7	94	4			
original 9	255	248	7	102	4	2	4	0
derived 9	253		7	98	4			
original 10	254	252	7	98	4	0	0	0
derived 10	254		7	98	4			

Fig.8.4. Summary of experimental results for n=8.

n=9	Linear Complexity	LCS	Algebraic Degree	Non Linearity	Algebraic Immunity	Δ Linear Complexity	Δ Non linearity	Δ Algebraic Immunity
original 1	510	508	8	206	4	1	4	0
derived 1	509		8	202	4			
original 2	509	504	8	214	4	2	0	0
derived 2	511		8	214	4			
original 3	511	510	8	210	4	0	0	1
derived 3	511		8	210	5			
original 4	511	504	8	214	5	2	0	0
derived 4	509		8	214	5			
original 5	511	506	8	210	4	0	0	0
derived 5	511		8	210	4			
original 6	509	510	8	206	4	2	0	0
derived 6	511		8	206	4			
original 7	511	506	8	206	5	1	4	0
derived 7	510		8	210	5			
original 8	509	508	8	210	5	1	0	0
derived 8	510		8	210	5			
original 9	511	506	8	206	4	0	2	0
derived 9	511		8	204	4			
original 10	509	508	8	206	4	2	0	0
derived 10	511		8	206	4			

Fig.8.5. Summary of experimental results for n=9.

1. The process /algorithm for deriving De Bruijn sequences from existing ones is proven from the results quite successful or at least it produces interesting results.

2. It doesn't always derive the ideal one e.g. with LCS = 2^n-2 however is produces optimal sequences nearby with very good rates e.g.:

2.1. For n=5, 2^n-2 =80%, 2^n-4 =10% , 2^n-6 =10%,
2.2. For n =6, 2^n-2 =0%, 2^n-4 =60% , 2^n-6 =40%,
2.3. For n=7, 2^n-2 =40%, 2^n-4 =30% , 2^n-6 =20%, 2^n-6 =10%,
2.4. For n=8, 2^n-2 =50%, 2^n-4 =20% , 2^n-6 =20%, 2^n-6 =10%,
2.5. For n=9, 2^n-2 =20%, 2^n-4 =30% , 2^n-6 =30%, 2^n-6 =20%,

OR more pictorial:

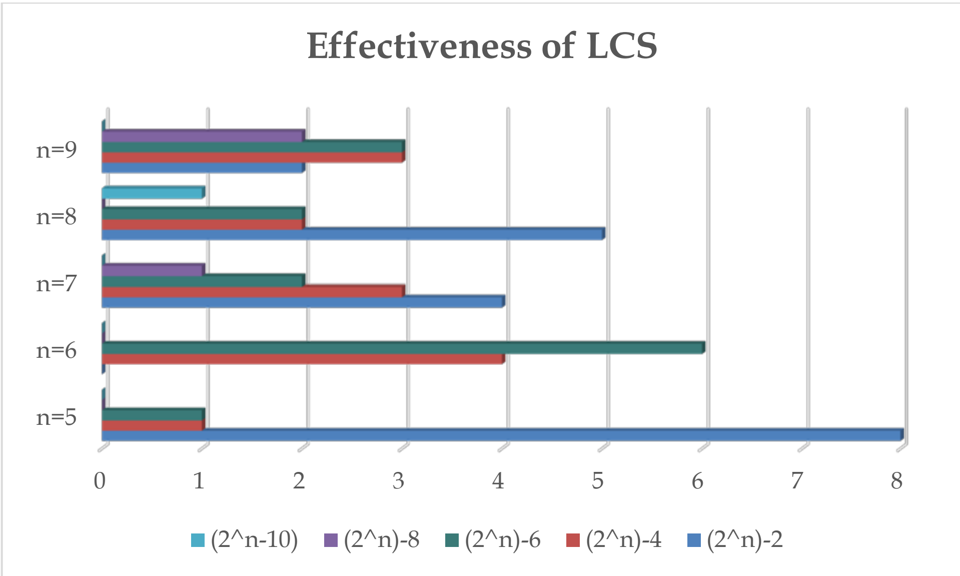


Fig. 8.6. Table with showing the effectiveness in achieving the Longer Common Subsequence.

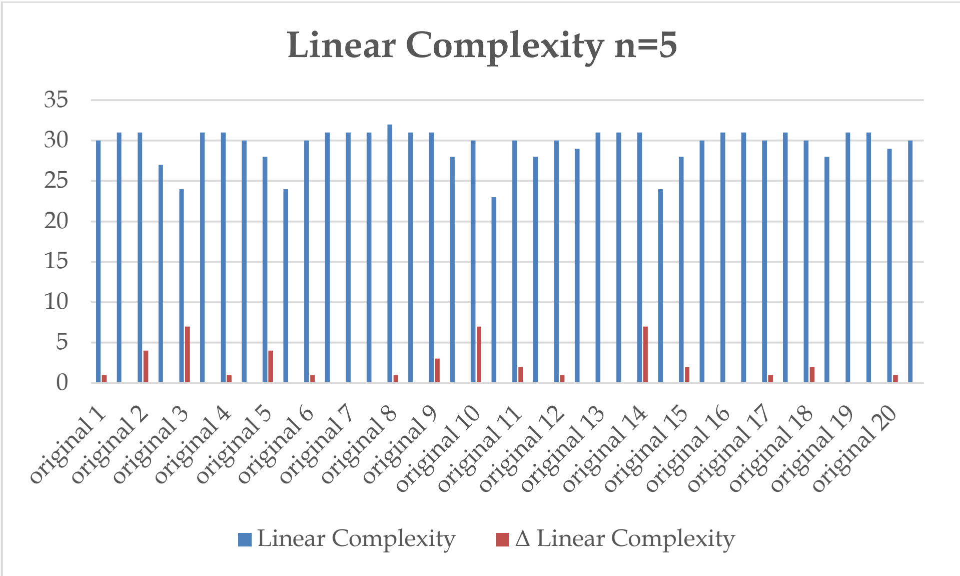
3. Cryptographic properties of relevant De Bruijn generators of the primary and the derived De Bruijn sequences present the following patterns:

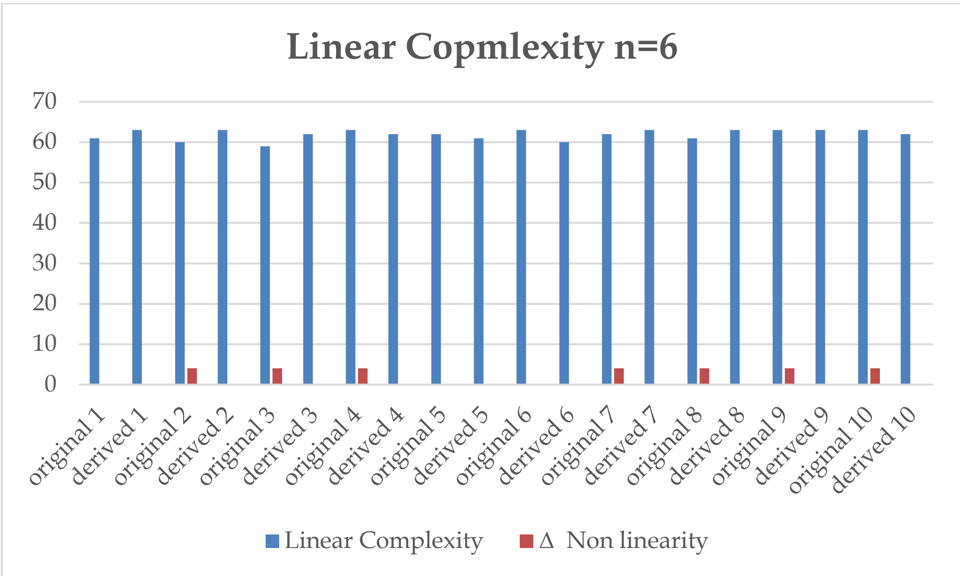
3.1. **Linear complexity:**

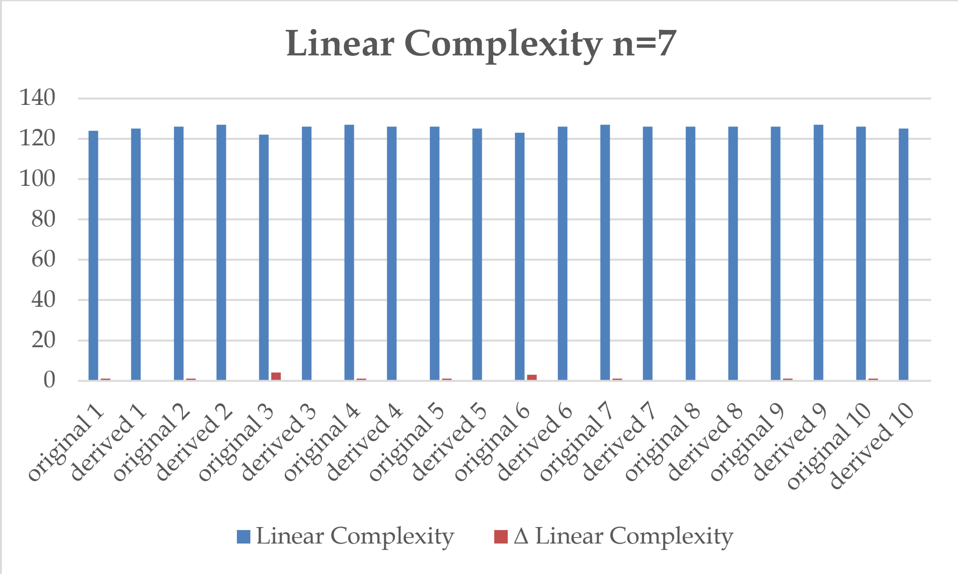
The chart below depicts linear-complexity resulting from the implementation of the algorithm as a function of the primary and the derived sequences together with the delta function Δ.

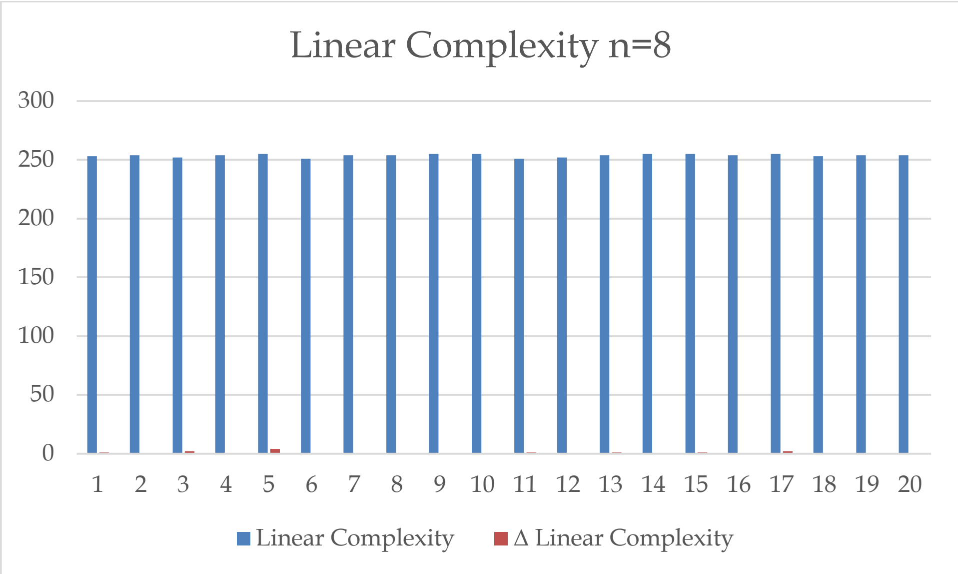
The charts do not present any particular trend. On the contrary, they appeared to be random with positive and negative variations without any justification.

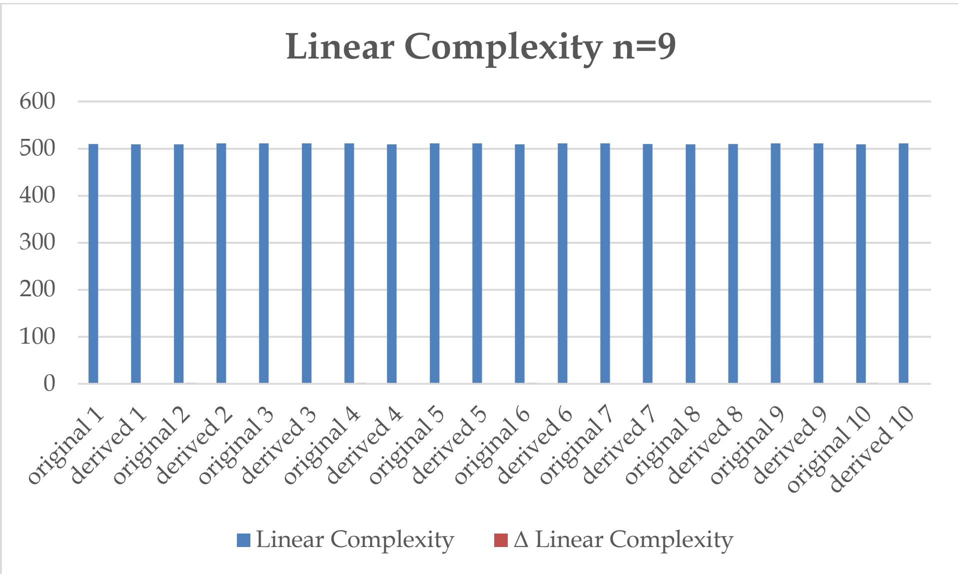
However, the exercise provides the opportunity to derive and select the sequence with the stronger linear-complexity degree to withstand the algebraic attacks.






Linear Complexity n=7
140
120
100
80
60
40
20
0
original 1
derived 1
original 2
derived 2
original 3
derived 3
original 4
derived 4
original 5
derived 5
original 6
derived 6
original 7
derived 7
original 8
derived 8
original 9
derived 9
original 10
derived 10
Linear Complexity
Δ Linear Complexity


Linear Complexity n=8
300
250
200
150
100
50
0
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Linear Complexity
Δ Linear Complexity


Linear Complexity n=9
600
500
400
300
200
100
0
original 1
derived 1
original 2
derived 2
original 3
derived 3
original 4
derived 4
original 5
derived 5
original 6
derived 6
original 7
derived 7
original 8
derived 8
original 9
derived 9
original 10
derived 10
Linear Complexity
Δ Linear Complexity

4. **Cryptographic properties of the corresponding Boolean functions**.

4.1. For each pair of sequences that share the LCS, both their linear complexity and the cryptographic properties of the corresponding Boolean functions that produce them are the same or very close. This is considered normal at first since reference is made to very similar sequences – and so, are the corresponding Boolean functions.

4.2. Examining the tables of fig.8.2 -fig.8.5 and particularly the Δ columns it can be seen that the differences between the measured crypto properties as very small. However, even the small differences encountered in some cases are not necessarily negligible.

4.3. The table below shows the maximum nonlinearity and the maximum possible algebraic immunity that any equal Boolean function with n variables best known VS Max and Min achieved in the exercise.

n	Best known nonlinearity for balanced function with n variablesVS Max/Min Produced			Maximum possible algebraic immunity for any function with n variables VS Max/Min Produced		
	Best Known	Max achieved	Min achieved	Best Known	Max achieved	Min achieved
5	12	10	6	3	3	2
6	26	22	14	3	3	3
7	56	50	38	4	3	3
8	116	102	90	4	4	4
9	240	214	202	5	5	4

Comparing the above results in terms of,

no-linearity, the values achieved are far from the best known (maximum possible) while in terms of Algebraic Immunity relatively closed. The difference showed the Δ column may be considered negligible, but when already the achieved values are far from the ideal (best known) any further reduction is not desirable. The lower the nonlinearity the higher the risk to have successful linear approximation attacks.

algebraic Immunity, the deviation is small, less than a unit. However, even small they have a considerable significance which is being proven that a 1% reduction in algebraic immunity can play a major role in the susceptance to an algebraic attack.

Therefore, if it is to have a primary De Bruijn sequence generator in a cryptographic algorithm due to the known advantages e.g. max period, goes through all the states, etc, then the construction of such a generator shall be very similar to its derived sequence.

To this effect, it is desirable that the derived sequence with LCS then is important that the newly created must have equivalent cryptographic criteria. Otherwise, an LCS with weaker Boolean cryptographic parameters could provide the opportunity to the potential attacker for an approximation attack.

4.4. **Algebraic Degree.**

The algebraic degree remains constant for all the sequences in the group irrespective of whether is original or derived. It appears that the Algebraic Degree to always be at n-1 with zero variation.

Generators based on De Bruijn sequences with a low algebraic degree of the Boolean function that produces it is prone to Interpolation attacks.

In our case, since the Algebraic Degree remains constant for all constructions their crypto properties are solely dependent on the n variables that create them.

Finally, it must be stated that the study has been developed using visual records techniques. The same can be reproduced using program coding and structured data or databases.

Both methods can be used to investigate further the De Bruijn sequences and their LCS. The method with the visual records operates on a record by record basis in a manual way and the handling of the matrices becomes very difficult as the size of the sequence increase. However, it provides the opportunity to direct the process having visibility on the next steps.

Employing computer programming and data structures is certainly more automated and can produce faster results and handled linger sequences. However, the complication in the selection of the pairs could make the process more random depending on the conditions predicted/include in the program.

Conclusions:

This thesis studies binary De Bruijn sequences, in terms of providing some new results that could be of cryptographic importance. More precisely, the basic question was to find an algorithm that given a binary De Bruijn sequence y, computes another binary De Bruijn sequence y ’with the largest common subsequence with the initial one. This problem was first studied very recently, but no such algorithm is known. In this work, via exploiting the so-called inverse suffix arrays of De Bruijn sequences, we developed a new approximation algorithm for this problem.

The proposed methodology and the corresponding test results are proven to be successful. In many cases, the algorithm manages to find a sequence that is indeed the "best approximating" to the original one. In all cases though, good approximations (possibly best) are efficiently computed (a brute force search over all possible cross-join pairs is needed to check the optimality, which is of high computational cost for large De Bruijn sequences).

Having such an algorithm as a tool, we subsequently examined the cryptographic criteria (namely, the algebraic degree, nonlinearity, and algebraic immunity) of the corresponding Boolean functions that generate such “highly similar” De Bruin sequences; in this context, their linear complexities were also examined to evaluate their behavior. The main outcomes of our experimental results rest with the fact the main cryptographic criteria remain, in general, invariant (something that is not surprising). However, there exist cases in which the nonlinearity seems to behave differently for such “highly similar” De Bruijn sequences, whereas some small variations also appear for the algebraic immunity. Hence, it is of cryptographic importance to check the properties of cryptographic Boolean functions that generate similar De Bruijn sequences.

Therefore, from this study, a new cryptographic criterion of De Bruijn generators naturally turns out: it is essential not only to ensure that this generator – as a Boolean function – possesses strong cryptographic criteria, but also the same should hold for the Boolean function generating another De Bruijn sequence with the longest common subsequence with the original one. If one of them does not behave well in terms of one criterion, then any of these two De Bruijn generators should be better excluded in the context of developing strong cryptographic primitives.

Future research could use the proposed methodology and platform for verifying the strength of the parameters of known cryptographic registers. Moreover, as an important open research question, we state the need to evaluate how exact are the solutions of our proposed approximation algorithm

BIBLIOGRAPHY

[1] J. Yupeng and L. Dongdai, "Longest subsequences shared by two de Bruijn sequences," Springer Science+Business Media, LLC, part of Springer Nature 2020, 2020.

[2] N. K. a. D. K. Konstantinos Limniotis, "De Bruijn Sequences and Suffix Arrays:," in *Modern Discrete Mathematics and Analysis*, Springer International Publishing AG, part of Springer Nature 2018, 2018, pp. 298-316.

[3] d. N. G. Bruijn, "A Combinatorial Problem," in *Proceedings of the Section of Sciences of the Koninklijke*, Amsterdam, 1946..

[4] T. L. A. Etzion, "Construction of de Bruijn sequences of minimal complexity," *IEEE,* pp. 705-708, 1984.

[5] H. C. J. S. van Tilborg, Encyclopedia of Cryptography and Security, Boston: Springer, Boston, MA, 2011.

[6] U. M. G. Manber, "Suffix Arrays: A New Method for On-Line searches.," *Society for Industrial and Applied Mathematics,* vol. 22, no. 5, p. 935–948, October 1993.

[7] F. S. N. MacWilliams, The Theory of Error-Correcting Codes, Amsterdam: North-Holland, 1977.

[8] C. Limniotis, *Lecture Notes: AYD 621 Cryptography,* Nicosia: OUC, 2020 -2021.

[9] S. J. Mykkeltveit J., "On cross joining de Bruijn sequences," *Am. Math. Soc.,* no. 632, p. 335–346, 2015.

[11] G. Marsaglia, "A Current View of Random Number Generators," in *Computer Science and Statistics, Sixteenth Symposium on the Interface*, Amsterdam, 1985.

[12] J. S. M. T. T. Mykkeltveit, "On the cycle structure of some nonlinear shift register," *Inform. Control 43,* pp. 202-215, 1979.

[13] T. L. G. A. H. A. S. P. K. Kasai, "Linear-time longest-common prefix computation in suffix arrays and its applications.," Springer, Heidelberg, 2001.

Appendix A
Binary and ISA form of De Bruijn sequences

A.1. De Bruin Sequences of order n=5.

00000111110011000101011101101001
0 1 3 7 15 31 30 28 25 19 6 12 24 17 2 5 10 21 11 23 14 29 27 22 13 26 20 9 18 4 8 16
00000101011111000111011001001101
0 1 2 5 10 21 11 23 15 31 30 28 24 17 3 7 14 29 27 22 12 25 18 4 9 19 6 13 26 20 8 16
00000110001010011101011011111001
0 1 3 6 12 24 17 2 5 10 20 9 19 7 14 29 26 21 11 22 13 27 23 15 31 30 28 25 18 4 8 16
00000100011001011010100111011111
0 1 2 4 8 17 3 6 12 25 18 5 11 22 13 26 21 10 20 9 19 7 14 29 27 23 15 31 30 28 24 16
00000111001000101111101010011011
0 1 3 7 14 28 25 18 4 8 17 2 5 11 23 15 31 30 29 26 21 10 20 9 19 6 13 27 22 12 24 16
00000100101101010001110111110011
0 1 2 4 9 18 5 11 22 13 26 21 10 20 8 17 3 7 14 29 27 23 15 31 30 28 25 19 6 12 24 16
00000111000100101011001101111101
0 1 3 7 14 28 24 17 2 4 9 18 5 10 21 11 22 12 25 19 6 13 27 23 15 31 30 29 26 20 8 16
00000111110111000100101100110101
0 1 3 7 15 31 30 29 27 23 14 28 24 17 2 4 9 18 5 11 22 12 25 19 6 13 26 21 10 20 8 16
00000101011111000110110011101001
0 1 2 5 10 21 11 23 15 31 30 28 24 17 3 6 13 27 22 12 25 19 7 14 29 26 20 9 18 4 8 16

00000110101110110001010011111001
0 1 3 6 13 26 21 11 23 14 29 27 22 12 24 17 2 5 10 20 9 19 7 15 31 30 28 25 18 4 8 16
00000101011111001100011101101001
0 1 2 5 10 21 11 23 15 31 30 28 25 19 6 12 24 17 3 7 14 29 27 22 13 26 20 9 18 4 8 16
00000101101111100100011101010011
0 1 2 5 11 22 13 27 23 15 31 30 28 25 18 4 8 17 3 7 14 29 26 21 10 20 9 19 6 12 24 16
00000100110101100011101111100101
0 1 2 4 9 19 6 13 26 21 11 22 12 24 17 3 7 14 29 27 23 15 31 30 28 25 18 5 10 20 8 16
00000100011001111101001010110111
0 1 2 4 8 17 3 6 12 25 19 7 15 31 30 29 26 20 9 18 5 10 21 11 22 13 27 23 14 28 24 16
00000110111010001010110010011111
0 1 3 6 13 27 23 14 29 26 20 8 17 2 5 10 21 11 22 12 25 18 4 9 19 7 15 31 30 28 24 16
00000110101000100111110111001011
0 1 3 6 13 26 21 10 20 8 17 2 4 9 19 7 15 31 30 29 27 23 14 28 25 18 5 11 22 12 24 16
00000110010101101111101000100111
0 1 3 6 12 25 18 5 10 21 11 22 13 27 23 15 31 30 29 26 20 8 17 2 4 9 19 7 14 28 24 16
00000100101101110001100111110101
0 1 2 4 9 18 5 11 22 13 27 23 14 28 24 17 3 6 12 25 19 7 15 31 30 29 26 21 10 20 8 16
00000101101111100011101010011001
0 1 2 5 11 22 13 27 23 15 31 30 28 24 17 3 7 14 29 26 21 10 20 9 19 6 12 25 18 4 8 16
00000111110100010101101110010011
0 1 3 7 15 31 30 29 26 20 8 17 2 5 10 21 11 22 13 27 23 14 28 25 18 4 9 19 6 12 24 16

A.3. De Bruin Sequences of order n=6.

0000001010100001101111110001111010010001001100101110011101101011
0 1 2 5 10 21 42 20 40 16 33 3 6 13 27 55 47 31 63 62 60 56 49 35 7 15 30 61 58 52 41 18 36 8 17 34 4 9 19 38 12 25 50 37 11 23 46 28 57 51 39 14 29 59 54 45 26 53 43 22 44 24 48 32
0000001010011001111010101101111110010010111011000100011010000111
0 1 2 5 10 20 41 19 38 12 25 51 39 15 30 61 58 53 42 21 43 22 45 27 55 47 31 63 62 60 57 50 36 9 18 37 11 23 46 29 59 54 44 24 49 34 4 8 17 35 6 13 26 52 40 16 33 3 7 14 28 56 48 32
0000001001011001111110001010011010000111011011110101011100100011
0 1 2 4 9 18 37 11 22 44 25 51 39 15 31 63 62 60 56 49 34 5 10 20 41 19 38 13 26 52 40 16 33 3 7 14 29 59 54 45 27 55 47 30 61 58 53 42 21 43 23 46 28 57 50 36 8 17 35 6 12 24 48 32
0000001011010100001100101011111101100011100110111010010001001111
0 1 2 5 11 22 45 26 53 42 20 40 16 33 3 6 12 25 50 37 10 21 43 23 47 31 63 62 61 59 54 44 24 49 35 7 14 28 57 51 38 13 27 55 46 29 58 52 41 18 36 8 17 34 4 9 19 39 15 30 60 56 48 32
0000001001100111111011100011010010000111010101111001010001011011
0 1 2 4 9 19 38 12 25 51 39 15 31 63 62 61 59 55 46 28 56 49 35 6 13 26 52 41 18 36 8 16 33 3 7 14 29 58 53 42 21 43 23 47 30 60 57 50 37 10 20 40 17 34 5 11 22 45 27 54 44 24 48 32
0000001001000101011100111010000111111000110010110101001101111011
0 1 2 4 9 18 36 8 17 34 5 10 21 43 23 46 28 57 51 39 14 29 58 52 40 16 33 3 7 15 31 63 62 60 56 49 35 6 12 25 50 37 11 22 45 26 53 42 20 41 19 38 13 27 55 47 30 61 59 54 44 24 48 32
0000001001111010111001010100011011010010000111011111100010110011
0 1 2 4 9 19 39 15 30 61 58 53 43 23 46 28 57 50 37 10 21 42 20 40 17 35 6 13 27 54 45 26 52 41 18 36 8 16 33 3 7 14 29 59 55 47 31 63 62 60 56 49 34 5 11 22 44 25 51 38 12 24 48 32
0000001100100001001110011011101010110001011110110100101000111111

0 1 3 6 12 25 50 36 8 16 33 2 4 9 19 39 14 28 57 51 38 13 27 55 46 29 58 53 42 21 43 22 44 24 49 34 5 11
23 47 30 61 59 54 45 26 52 41 18 37 10 20 40 17 35 7 15 31 63 62 60 56 48 32
0000011100101101110110011000100001010010011110100011010101111111
0 1 3 7 14 28 57 50 37 11 22 45 27 55 46 29 59 54 44 25 51 38 12 24 49 34 4 8 16 33 2 5 10 20 41 18 36 9
19 39 15 30 61 58 52 40 17 35 6 13 26 53 42 21 43 23 47 31 63 62 60 56 48 32
0000001001000110010100111111011011110011010000111000101110101011
0 1 2 4 9 18 36 8 17 35 6 12 25 50 37 10 20 41 19 39 15 31 63 62 61 59 54 45 27 55 47 30 60 57 51 38 13
26 52 40 16 33 3 7 14 28 56 49 34 5 11 23 46 29 58 53 42 21 43 22 44 24 48 32

A.4. De Bruin Sequences of order n=7.

0000000101101000000110001000111101010010011111001110010100010101011101111000110111000011
1010011001000010010111111101101100110 1011
0 1 2 5 11 22 45 90 52 104 80 32 65 3 6 12 24 49 98 68 8 17 35 71 15 30 61 122 117 106 84 41 82 36 73 19
39 79 31 62 124 121 115 103 78 28 57 114 101 74 20 40 81 34 69 10 21 42 85 43 87 46 93 59 119 111 94
60 120 113 99 70 13 27 55 110 92 56 112 97 67 7 14 29 58 116 105 83 38 76 25 50 100 72 16 33 66 4 9 18
37 75 23 47 95 63 127 126 125 123 118 109 91 54 108 89 51 102 77 26 53 107 86 44 88 48 96 64
0000000111001000001100111101101110100110000010001001110111111010100011111110011011000110 1
0111100010110010100100101110000101010 1101
0 1 3 7 14 28 57 114 100 72 16 33 67 6 12 25 51 103 79 30 61 123 118 109 91 55 110 93 58 116 105 83 38
76 24 48 96 65 2 4 8 17 34 68 9 19 39 78 29 59 119 111 95 62 125 122 117 106 84 40 81 35 71 15 31 63
127 126 124 121 115 102 77 27 54 108 88 49 99 70 13 26 53 107 87 47 94 60 120 113 98 69 11 22 44 89 50
101 74 20 41 82 36 73 18 37 75 23 46 92 56 112 97 66 5 10 21 42 85 43 86 45 90 52 104 80 32 64
0000000111101011100011001010110011111011101001011110010000110111111110000101001100010110
1100000100011101101010100010010011100 1101
0 1 3 7 15 30 61 122 117 107 87 46 92 56 113 99 70 12 25 50 101 74 21 43 86 44 89 51 103 79 31 62 125
123 119 110 93 58 116 105 82 37 75 23 47 94 60 121 114 100 72 16 33 67 6 13 27 55 111 95 63 127 126
124 120 112 97 66 5 10 20 41 83 38 76 24 49 98 69 11 22 45 91 54 108 88 48 96 65 2 4 8 17 35 71 14 29 59
118 109 90 53 106 85 42 84 40 81 34 68 9 18 36 73 19 39 78 28 57 115 102 77 26 52 104 80 32 64
0000000111111010010001001010000100000101011001001111011101011110001011010101001100001101
0001110110001100111001101101111111100 10111
0 1 3 7 15 31 62 125 122 116 105 82 36 72 17 34 68 9 18 37 74 20 40 80 33 66 4 8 16 32 65 2 5 10 21 43 86
44 89 50 100 73 19 39 79 30 61 123 119 110 93 58 117 107 87 47 94 60 120 113 98 69 11 22 45 90 53 106
85 42 84 41 83 38 76 24 48 97 67 6 13 26 52 104 81 35 71 14 29 59 118 108 88 49 99 70 12 25 51 103 78
28 57 115 102 77 27 54 109 91 55 111 95 63 127 126 124 121 114 101 75 23 46 92 56 112 96 64
0000000110010110011110100010000100101010110111110000010111000111010111100110000111111110
1110110101001001110010001101001101100 0101
0 1 3 6 12 25 50 101 75 22 44 89 51 103 79 30 61 122 116 104 81 34 68 8 16 33 66 4 9 18 37 74 21 42 85
43 86 45 91 55 111 95 62 124 120 112 96 65 2 5 11 23 46 92 56 113 99 71 14 29 58 117 107 87 47 94 60
121 115 102 76 24 48 97 67 7 15 31 63 127 126 125 123 119 110 93 59 118 109 90 53 106 84 41 82 36 73
19 39 78 28 57 114 100 72 17 35 70 13 26 52 105 83 38 77 27 54 108 88 49 98 69 10 20 40 80 32 64

0000000101100110111011000100101000010001111010011000011100111010100100000110100010101011011010111000110010111100100111111011111
0 1 2 5 11 22 44 89 51 102 77 27 55 110 93 59 118 108 88 49 98 68 9 18 37 74 20 40 80 33 66 4 8 17 35 71 15 30 61 122 116 105 83 38 76 24 48 97 67 7 14 28 57 115 103 78 29 58 117 106 84 41 82 36 72 16 32 65 3 6 13 26 52 104 81 34 69 10 21 42 85 43 86 45 91 54 109 90 53 107 87 46 92 56 113 99 70 12 25 50 101 75 23 47 94 60 121 114 100 73 19 39 79 31 63 127 126 125 123 119 111 95 62 124 120 112 96 64
00000001100001011110100011110010100000100001110101110011011000110101010010110010011100010010001010110100110011111011011101111111
0 1 3 6 12 24 48 97 66 5 11 23 47 94 61 122 116 104 81 35 71 15 30 60 121 114 101 74 20 40 80 32 65 2 4 8 16 33 67 7 14 29 58 117 107 87 46 92 57 115 102 77 27 54 108 88 49 99 70 13 26 53 106 85 42 84 41 82 37 75 22 44 89 50 100 73 19 39 78 28 56 113 98 68 9 18 36 72 17 34 69 10 21 43 86 45 90 52 105 83 38 76 25 51 103 79 31 62 125 123 118 109 91 55 110 93 59 119 111 95 63 127 126 124 120 112 96 64
00000001011111000101011000110110111100111001010010110101010000110011010001000000111101100100011101110100110000100100111111010111
0 1 2 5 11 23 47 95 62 124 120 113 98 69 10 21 43 86 44 88 49 99 70 13 27 54 109 91 55 111 94 60 121 115 103 78 28 57 114 101 74 20 41 82 37 75 22 45 90 53 106 85 42 84 40 80 33 67 6 12 25 51 102 77 26 52 104 81 34 68 8 16 32 65 3 7 15 30 61 123 118 108 89 50 100 72 17 35 71 14 29 59 119 110 93 58 116 105 83 38 76 24 48 97 66 4 9 18 36 73 19 39 79 31 63 127 126 125 122 117 107 87 46 92 56 112 96 64
00000001100100111010011000001000110110001001000011110101010010111100011100001010001011001101011011100111111100101011101111101101
0 1 3 6 12 25 50 100 73 19 39 78 29 58 116 105 83 38 76 24 48 96 65 2 4 8 17 35 70 13 27 54 108 88 49 98 68 9 18 36 72 16 33 67 7 15 30 61 122 117 106 85 42 84 41 82 37 75 23 47 94 60 120 113 99 71 14 28 56 112 97 66 5 10 20 40 81 34 69 11 22 44 89 51 102 77 26 53 107 86 45 91 55 110 92 57 115 103 79 31 63 127 126 124 121 114 101 74 21 43 87 46 93 59 119 111 95 62 125 123 118 109 90 52 104 80 32 64
00000001000111101110000011100111010111111101000100100111110001100000110110111100100001011000101001010101101001100101110110011010 1
0 1 2 4 8 17 35 71 15 30 61 123 119 110 92 56 112 97 67 7 14 28 57 115 103 78 29 58 117 107 87 47 95 63 127 126 125 122 116 104 81 34 68 9 18 36 73 19 39 79 31 62 124 120 113 99 70 12 24 48 96 65 3 6 13 27 54 109 91 55 111 94 60 121 114 100 72 16 33 66 5 11 22 44 88 49 98 69 10 20 41 82 37 74 21 42 85 43 86 45 90 52 105 83 38 76 25 50 101 75 23 46 93 59 118 108 89 51 102 77 26 53 106 84 40 80 32 64

A.5. De Bruin Sequences of order n=8.

00000000101000110100100000011101001110001111010001011000000100111100101001100100101011010000011000110011101101101010100101111000011111111000101010111011111001111101110101000011011001100001011100110111101100010000100011100100010010011010111111010110010110111

0 1 2 5 10 20 40 81 163 70 141 26 52 105 210 164 72 144 32 64 129 3 7 14 29 58 116 233 211 167 78 156 56 113 227 199 143 30 61 122 244 232 209 162 69 139 22 44 88 176 96 193 130 4 9 19 39 79 158 60 121 242 229 202 148 41 83 166 76 153 50 100 201 146 37 74 149 43 86 173 90 180 104 208 160 65 131 6 12 24 49 99 198 140 25 51 103 206 157 59 118 237 219 182 109 218 181 106 213 170 84 169 82 165 75 151 47 94 188 120 240 225 195 135 15 31 63 127 255 254 252 248 241 226 197 138 21 42 85 171 87 174 93 187 119 239 223 190 124 249 243 231 207 159 62 125 251 247 238 221 186 117 234 212 168 80 161 67 134 13

27 54 108 217 179 102 204 152 48 97 194 133 11 23 46 92 185 115 230 205 155 55 111 222 189 123 246 236 216 177 98 196 136 16 33 66 132 8 17 35 71 142 28 57 114 228 200 145 34 68 137 18 36 73 147 38 77 154 53 107 215 175 95 191 126 253 250 245 235 214 172 89 178 101 203 150 45 91 183 110 220 184 112 224 192 128

000000001001010011100101011011100000101000111110000111100011101110100110101110011011111 010001001100011000000110011110101001001000010000011100010001101100101100001011101100010 110100001101001011110011101011001100100111111100100010101011111101111011011010101

0 1 2 4 9 18 37 74 148 41 83 167 78 156 57 114 229 202 149 43 86 173 91 183 110 220 184 112 224 193 130 5 10 20 40 81 163 71 143 31 62 124 248 240 225 195 135 15 30 60 120 241 227 199 142 29 59 119 238 221 186 116 233 211 166 77 154 53 107 215 174 92 185 115 230 205 155 55 111 223 190 125 250 244 232 209 162 68 137 19 38 76 152 49 99 198 140 24 48 96 192 129 3 6 12 25 51 103 207 158 61 122 245 234 212 169 82 164 73 146 36 72 144 33 66 132 8 16 32 65 131 7 14 28 56 113 226 196 136 17 35 70 141 27 54 108 217 178 101 203 150 44 88 176 97 194 133 11 23 46 93 187 118 236 216 177 98 197 139 22 45 90 180 104 208 161 67 134 13 26 52 105 210 165 75 151 47 94 188 121 243 231 206 157 58 117 235 214 172 89 179 102 204 153 50 100 201 147 39 79 159 63 127 255 254 252 249 242 228 200 145 34 69 138 21 42 85 171 87 175 95 191 126 253 251 247 239 222 189 123 246 237 219 182 109 218 181 106 213 170 84 168 80 160 64 128

000000001111001111011000110001000000100010010001110100100101101110101010110101110111110 000111000101010001101101000001010000101100001001101111000111110111001000011001110110110 010101111110010111000001101001100100111001101010010100111111110100010111101011 0011

0 1 3 7 15 30 60 121 243 231 207 158 61 123 246 236 216 177 99 198 140 24 49 98 196 136 16 32 64 129 2 4 8 17 34 68 137 18 36 72 145 35 71 142 29 58 116 233 210 164 73 146 37 75 150 45 91 183 110 221 186 117 234 213 170 85 171 86 173 90 181 107 215 174 93 187 119 239 223 190 124 248 240 225 195 135 14 28 56 113 226 197 138 21 42 84 168 81 163 70 141 27 54 109 218 180 104 208 160 65 130 5 10 20 40 80 161 66 133 11 22 44 88 176 97 194 132 9 19 38 77 155 55 111 222 188 120 241 227 199 143 31 62 125 251 247 238 220 185 114 228 200 144 33 67 134 12 25 51 103 206 157 59 118 237 219 182 108 217 178 101 202 149 43 87 175 95 191 126 252 249 242 229 203 151 46 92 184 112 224 193 131 6 13 26 52 105 211 166 76 153 50 100 201 147 39 78 156 57 115 230 205 154 53 106 212 169 82 165 74 148 41 83 167 79 159 63 127 255 254 253 250 244 232 209 162 69 139 23 47 94 189 122 245 235 214 172 89 179 102 204 152 48 96 192 128

000000001000010011110011000001100010001001011111101001110100010111010111001010010101100 110101101101010111100000011101100001101001001100100100000101101000011110110111110001010 101000110111011110101001101100101100011001110011111011100011111111001000111000 0101

0 1 2 4 8 16 33 66 132 9 19 39 79 158 60 121 243 230 204 152 48 96 193 131 6 12 24 49 98 196 136 17 34 68 137 18 37 75 151 47 95 191 126 253 250 244 233 211 167 78 157 58 116 232 209 162 69 139 23 46 93 186 117 235 215 174 92 185 114 229 202 148 41 82 165 74 149 43 86 172 89 179 102 205 154 53 107 214 173 91 182 109 218 181 106 213 171 87 175 94 188 120 240 224 192 129 3 7 14 29 59 118 236 216 176 97 195 134 13 26 52 105 210 164 73 147 38 76 153 50 100 201 146 36 72 144 32 65 130 5 11 22 45 90 180 104 208 161 67 135 15 30 61 123 246 237 219 183 111 223 190 124 248 241 226 197 138 21 42 85 170 84 168 81 163 70 141 27 55 110 221 187 119 239 222 189 122 245 234 212 169 83 166 77 155 54 108 217 178 101 203 150 44 88 177 99 198 140 25 51 103 206 156 57 115 231 207 159 62 125 251 247 238 220 184 113 227 199 143 31 63 127 255 254 252 249 242 228 200 145 35 71 142 28 56 112 225 194 133 10 20 40 80 160 64 128

000000001111010111101111111101001010001011111011000110001010011111000100010011101110110
100011100011111100101010010000010101110101000011010101011011011100111001001011000010110
1011001000110111100110110011110000001000010010011010011001100000110010111000011101

0 1 3 7 15 30 61 122 245 235 215 175 94 189 123 247 239 223 191 127 255 254 253 250 244 233 210 165 74 148 40 81 162 69 139 23 47 95 190 125 251 246 236 216 177 99 198 140 24 49 98 197 138 20 41 83 167 79 159 62 124 248 241 226 196 136 17 34 68 137 19 39 78 157 59 119 238 221 187 118 237 218 180 104 209 163 71 142 28 56 113 227 199 143 31 63 126 252 249 242 229 202 149 42 84 169 82 164 72 144 32 65 130 5 10 21 43 87 174 93 186 117 234 212 168 80 161 67 134 13 26 53 106 213 170 85 171 86 173 91 182 109 219 183 110 220 185 115 231 206 156 57 114 228 201 146 37 75 150 44 88 176 97 194 133 11 22 45 90 181 107 214 172 89 178 100 200 145 35 70 141 27 55 111 222 188 121 243 230 205 155 54 108 217 179 103 207 158 60 120 240 224 192 129 2 4 8 16 33 66 132 9 18 36 73 147 38 77 154 52 105 211 166 76 153 51 102 204 152 48 96 193 131 6 12 25 50 101 203 151 46 92 184 112 225 195 135 14 29 58 116 232 208 160 64 128

000000001011000010100111000011111111011010000100011101001000001110010001010111011111100
000011000001001111001011100111011000100101101010001100110111101110101111100110010010011
0001111010101011011011100010111100011011001111101000100001101001101011001010100101

0 1 2 5 11 22 44 88 176 97 194 133 10 20 41 83 167 78 156 56 112 225 195 135 15 31 63 127 255 254 253 251 246 237 218 180 104 208 161 66 132 8 17 35 71 142 29 58 116 233 210 164 72 144 32 65 131 7 14 28 57 114 228 200 145 34 69 138 21 43 87 174 93 187 119 239 223 191 126 252 248 240 224 192 129 3 6 12 24 48 96 193 130 4 9 19 39 79 158 60 121 242 229 203 151 46 92 185 115 231 206 157 59 118 236 216 177 98 196 137 18 37 75 150 45 90 181 106 212 168 81 163 70 140 25 51 102 205 155 55 111 222 189 123 247 238 221 186 117 235 215 175 95 190 124 249 243 230 204 153 50 100 201 146 36 73 147 38 76 152 49 99 199 143 30 61 122 245 234 213 170 85 171 86 173 91 182 109 219 183 110 220 184 113 226 197 139 23 47 94 188 120 241 227 198 141 27 54 108 217 179 103 207 159 62 125 250 244 232 209 162 68 136 16 33 67 134 13 26 52 105 211 166 77 154 53 107 214 172 89 178 101 202 149 42 84 169 82 165 74 148 40 80 160 64 128

000000001001101011000101111110000111110011011101111101010001101101001111111101100001000
111101000011001100011000000111000001101000100000101001100100100111010010000101101111000
1001010110101010111101110011110010001010100101100101110101110001110110110011100101

0 1 2 4 9 19 38 77 154 53 107 214 172 88 177 98 197 139 23 47 95 191 126 252 248 240 225 195 135 15 31 62 124 249 243 230 205 155 55 110 221 187 119 239 223 190 125 250 245 234 212 168 81 163 70 141 27 54 109 218 180 105 211 167 79 159 63 127 255 254 253 251 246 236 216 176 97 194 132 8 17 35 71 143 30 61 122 244 232 208 161 67 134 12 25 51 102 204 152 49 99 198 140 24 48 96 192 129 3 7 14 28 56 112 224 193 131 6 13 26 52 104 209 162 68 136 16 32 65 130 5 10 20 41 83 166 76 153 50 100 201 146 36 73 147 39 78 157 58 116 233 210 164 72 144 33 66 133 11 22 45 91 183 111 222 188 120 241 226 196 137 18 37 74 149 43 86 173 90 181 106 213 170 85 171 87 175 94 189 123 247 238 220 185 115 231 207 158 60 121 242 228 200 145 34 69 138 21 42 84 169 82 165 75 150 44 89 178 101 203 151 46 93 186 117 235 215 174 92 184 113 227 199 142 29 59 118 237 219 182 108 217 179 103 206 156 57 114 229 202 148 40 80 160 64 128

000000001001111011010000001100101110111001001100001110110000010101001110011011101011100
0010000011111101111100101011111111000111000101100010001010001100011011001100111010011010010
0001011111010101010110110111110011111110100010010010100101101010000110101100100011111

0 1 2 4 9 19 39 79 158 61 123 246 237 218 180 104 208 160 64 129 3 6 12 25 50 101 203 151 46 93 187 119 238 220 185 114 228 201 147 38 76 152 48 97 195 135 14 29 59 118 236 216 176 96 193 130 5 10 21 42 84 169 83 167 78 156 57 115 230 205 155 55 110 221 186 117 235 215 174 92 184 112 225 194 132 8

16 32 65 131 7 15 31 62 125 251 247 239 222 188 121 242 229 202 149 43 87 175 95 191 126 252 248 241 227 199 142 28 56 113 226 197 139 22 44 88 177 98 196 136 17 34 69 138 20 40 81 163 70 140 24 49 99 198 141 27 54 108 217 179 102 204 153 51 103 206 157 58 116 233 211 166 77 154 52 105 210 164 72 144 33 66 133 11 23 47 94 189 122 245 234 213 170 85 171 86 173 91 182 109 219 183 111 223 190 124 249 243 231 207 159 63 127 255 254 253 250 244 232 209 162 68 137 18 36 73 146 37 74 148 41 82 165 75 150 45 90 181 106 212 168 80 161 67 134 13 26 53 107 214 172 89 178 100 200 145 35 71 143 30 60 120 240 224 192 128

0000000010010101100000011011000101110111100001111110010110110101111101001011110110111111110111000100001000111101010010011111000111011001000100110010100010101011100110001100110101010000110000101001101110101101000000101100111100111010001101001110000011100100 1

0 1 2 4 9 18 37 74 149 43 86 172 88 176 96 192 129 3 6 13 27 54 108 216 177 98 197 139 23 46 93 187 119 239 222 188 120 240 225 195 135 15 31 63 126 252 249 242 229 203 150 45 91 182 109 218 181 107 215 175 95 190 125 250 244 233 210 165 75 151 47 94 189 123 246 237 219 183 111 223 191 127 255 254 253 251 247 238 220 184 113 226 196 136 16 33 66 132 8 17 35 71 143 30 61 122 245 234 212 169 82 164 73 147 39 79 159 62 124 248 241 227 199 142 29 59 118 236 217 178 100 200 145 34 68 137 19 38 76 153 50 101 202 148 40 81 162 69 138 21 42 85 171 87 174 92 185 115 230 204 152 49 99 198 140 25 51 102 205 154 53 106 213 170 84 168 80 161 67 134 12 24 48 97 194 133 10 20 41 83 166 77 155 55 110 221 186 117 235 214 173 90 180 104 208 160 65 130 5 11 22 44 89 179 103 207 158 60 121 243 231 206 157 58 116 232 209 163 70 141 26 52 105 211 167 78 156 56 112 224 193 131 7 14 28 57 114 228 201 146 36 72 144 32 64 128

0000000010000000110000010110100001101010001111100100111010101111010001011111010111010011110000111101100101100110110101100010101101100001010000011101111001100100011001111110111000110111011011111111000100010010010111001110010101010010100110100100001001100011 1

0 1 2 4 8 16 32 64 129 3 6 12 24 48 96 193 130 5 11 22 45 90 180 104 208 161 67 134 13 26 53 106 212 168 81 163 71 143 31 62 124 249 242 228 201 147 39 78 157 58 117 234 213 171 87 175 94 189 122 244 232 209 162 69 139 23 47 95 190 125 250 245 235 215 174 93 186 116 233 211 167 79 158 60 120 240 225 195 135 15 30 61 123 246 236 217 178 101 203 150 44 89 179 102 205 155 54 109 218 181 107 214 172 88 177 98 197 138 21 43 86 173 91 182 108 216 176 97 194 133 10 20 40 80 160 65 131 7 14 29 59 119 239 222 188 121 243 230 204 153 50 100 200 145 35 70 140 25 51 103 207 159 63 126 253 251 247 238 220 184 113 227 198 141 27 55 110 221 187 118 237 219 183 111 223 191 127 255 254 252 248 241 226 196 136 17 34 68 137 18 36 73 146 37 75 151 46 92 185 115 231 206 156 57 114 229 202 149 42 85 170 84 169 82 165 74 148 41 83 166 77 154 52 105 210 164 72 144 33 66 132 9 19 38 76 152 49 99 199 142 28 56 112 22

A.6. De Bruin Sequences of order n=9.

000000000100100001100001001100000010110100101100111110110101000100010100011100100101010010001001011100101111111010111011001000001101101111110111110010101100101000010111010101101011000011100001010101011100011001110011011000101011110101001110001001110110110100011000100000000110010011010111110100110011000111110000011101011011101110011101001001001111000011010011111111100111101110100000100001000110111000000011110100010111100100011110011001011011001101000011111100010110001101010100000001010010100110111100011101111011

0 1 2 4 9 18 36 72 144 289 67 134 268 24 48 97 194 388 265 19 38 76 152 304 96 193 386 261 11 22 45 90 180 361 210 421 331 150 300 89 179 359 207 415 318 125 251 502 493 474 437 362 212 424 337 162 324 136 273 34 69 138 276 40 81 163 327 142 284 57 114 228 457 402 293 74 149 298 84 169 338 164 328 145 290 68 137 274 37 75 151 302 92 185 370 229 459 407 303 95 191 383 254 509 506 501 491 471 430 349 187 374 236 473 434 356 200 400 288 65 131 262 13 27 54 109 219 439 367 223 447 382 253 507 503 495 479 446 380 249 498 485 458 405 299 86 172 345 178 357 202 404 296 80 161 322 133 267 23 46 93 186 373 234 469 427 342 173 346 181 363 214 428 344 176 353 195 391 270 28 56 112 225 450 389 266 21 42 85 170 341 171 343 174 348 184 369 227 454 396 281 51 103 206 412 313 115 230 461 411 310 108 216 433 354 197 394 277 43 87 175 350 189 378 245 490 468 425 339 167 334 156 312 113 226 452 393 275 39 78 157 315 118 237 475 438 365 218 436 360 209 419 326 140 280 49 98 196 392 272 32 64 128 257 3 6 12 25 50 100 201 403 294 77 154 309 107 215 431 351 190 381 250 500 489 467 422 332 153 307 102 204 408 305 99 199 399 287 62 124 248 496 480 449 387 263 14 29 58 117 235 470 429 347 183 366 221 443 375 238 476 441 371 231 462 413 314 116 233 466 420 329 146 292 73 147 295 79 158 316 120 240 481 451 390 269 26 52 105 211 423 335 159 319 127 255 511 510 508 505 499 487 463 414 317 123 247 494 477 442 372 232 464 416 321 130 260 8 16 33 66 132 264 17 35 70 141 283 55 110 220 440 368 224 448 385 259 7 15 30 61 122 244 488 465 418 325 139 279 47 94 188 377 242 484 456 401 291 71 143 286 60 121 243 486 460 409 306 101 203 406 301 91 182 364 217 435 358 205 410 308 104 208 417 323 135 271 31 63 126 252 504 497 482 453 395 278 44 88 177 355 198 397 282 53 106 213 426 340 168 336 160 320 129 258 5 10 20 41 82 165 330 148 297 83 166 333 155 311 111 222 444 376 241 483 455 398 285 59 119 239 478 445 379 246 492 472 432 352 192 384 256

00000000010111001001110011010001101001000001100101011101000001001100001101011010101011011 1101110101111111011011101110000101011000111110001110110100110111001110001100011011000101 1000111101010101000010001010100110010000110000010100111110111110101100111010011101111000 0001111000100011100000011100101100101111010001011000010010111011001101101011100010111110 0110001001001010101111001111111110000111010100010000101101101100100100010011110010100000 00010000000110111111010010100100110101001011010000111111001000110011001111011

0 1 2 5 11 23 46 92 185 370 228 457 403 295 78 156 313 115 230 461 410 308 104 209 419 326 141 282 52 105 210 420 328 144 288 65 131 262 12 25 50 101 202 405 299 87 174 349 186 372 232 464 416 321 130 260 9 19 38 76 152 304 97 195 390 269 26 53 107 214 429 346 181 362 213 427 342 173 347 183 367 222 445 379 247 494 477 442 373 235 471 431 351 191 383 254 509 507 502 493 475 439 366 221 443 375 238 476 440 368 225 450 389 266 21 43 86 172 344 177 355 199 399 287 62 124 248 497 483 455 398 285 59 118 237 474 436 361 211 422 333 155 311 110 220 441 371 231 462 412 312 113 227 454 396 280 49 99 198 397 283 54 108 216 433 354 197 394 276 40 81 163 327 143 286 61 122 245 490 469 426 341 170 340 168 336 161 322 132 264 17 34 69 138 277 42 84 169 339 166 332 153 306 100 200 400 289 67 134 268 24 48 96 193 386 261 10 20 41 83 167 335 159 318 125 251 503 495 479 446 381 250 501 491 470 428 345 179 359 206 413 314 116 233 467 423 334 157 315 119 239 478 444 376 240 480 449 387 263 15 30 60 120 241 482 452 392 273 35 71 142 284 56 112 224 448 385 259 7 14 28 57 114 229 459 406 300 89 178 357 203 407 303 94 189 378 244 488 465 418 325 139 278 44 88 176 353 194 388 265 18 37 75 151 302 93 187 374 236 473 435 358 205 411 310 109 218 437 363 215 430 348 184 369 226 453 395 279 47 95 190 380 249 499 486 460 408 305 98 196 393 274 36 73 146 293 74 149 298 85 171 343 175 350 188 377 243 487 463 415 319 127 255 511 510 508 504 496 481 451 391 270 29 58 117 234 468 424 337 162 324 136 272 33 66 133 267 22 45 91 182 365 219 438 364 217 434 356 201 402 292 72 145 290 68 137 275 39 79 158 316 121 242 485 458 404 296 80 160 320 129 258 4 8 16 32 64 128 257 3 6 13 27 55 111 223 447 382 253 506 500 489 466 421 330 148 297 82 164 329 147 294 77 154 309 106 212 425 338 165 331 150 301 90 180 360 208 417 323 135 271 31 63 126 252 505 498 484 456 401 291 70 140 281 51 102 204 409 307 103 207 414 317 123 246 492 472 432 352 192 384 256

000000000110101101000100011001110110010000001111000110100000001000100101100011000111101
101110111010100010111100101101010010111111010011011001111100001110100101000010000101000
111011110111001010010000110110110001001111111011000001010110000110010101011101101011110
100011011111001000111111001100001011100111100111001001000101001100010110111100000110000
001011001100110100111101011001011101000011111010101010011101011100010101000001110011011
100001001101010110110100100101011111011111111100010000010010011001001110001111

0 1 3 6 13 26 53 107 214 429 346 180 360 209 418 324 136 273 35 70 140 281 51 103 206 413 315 118 236 473 434 356 200 400 288 64 129 259 7 15 30 60 120 241 483 454 397 282 52 104 208 416 320 128 257 2 4 8 17 34 68 137 274 37 75 150 300 88 177 355 198 396 280 49 99 199 399 286 61 123 246 493 475 439 366 221 443 375 238 477 442 373 234 468 424 337 162 325 139 279 47 94 188 377 242 485 459 406 301 90 181 362 212 425 338 165 331 151 303 95 191 382 253 506 500 489 467 422 333 155 310 108 217 435 359 207 415 318 124 248 496 481 451 391 270 29 58 116 233 466 421 330 148 296 80 161 322 132 264 16 33 66 133 266 20 40 81 163 327 142 285 59 119 239 478 445 379 247 494 476 441 370 229 458 404 297 82 164 328 144 289 67 134 269 27 54 109 219 438 364 216 433 354 196 393 275 39 79 159 319 127 254 509 507 502 492 472 432 352 193 386 261 10 21 43 86 172 344 176 353 195 390 268 25 50 101 202 405 298 85 171 343 174 349 187 374 237 474 437 363 215 431 350 189 378 244 488 465 419 326 141 283 55 111 223 446 380 249 498 484 456 401 291 71 143 287 63 126 252 505 499 486 460 408 304 97 194 389 267 23 46 92 185 371 231 463 414 316 121 243 487 462 412 313 114 228 457 402 292 72 145 290 69 138 276 41 83 166 332 152 305 98 197 395 278 45 91 183 367 222 444 376 240 480 449 387 262 12 24 48 96 192 385 258 5 11 22 44 89 179 358 204 409 307 102 205 410 308 105 211 423 335 158 317 122 245 491 470 428 345 178 357 203 407 302 93 186 372 232 464 417 323 135 271 31 62 125 250 501 490 469 426 341 170 340 169 339 167 334 157 314 117 235 471 430 348 184 369 226 453 394 277 42 84 168 336 160 321 131 263 14 28 57 115 230 461 411 311 110 220 440 368 225 450 388 265 19 38 77 154 309 106 213 427 342 173 347 182 365 218 436 361 210 420 329 146 293 74 149 299 87 175 351 190 381 251 503 495 479 447 383 255 511 510 508 504 497 482 452 392 272 32 65 130 260 9 18 36 73 147 294 76 153 306 100 201 403 295 78 156 312 113 227 455 398 284 56 112 224 448 384 256

000000000111001010100101000100000100111111000100010111000110001011001100011011001111101
111101001001110000001010101000000001000010100110000110000001111001011011111111100110011100
111011110011110000000101101001110100110111101110111000010111111010111110000111110010000
110110110111001001000110011010110101001111011001000100101011110101000110101011011000010
001110101100011100010100001001001011101100010011001010010000001101000011011101000101011
001011110001111111011010111010101011100110110100011110100000110010011010010 11

0 1 3 7 14 28 57 114 229 458 405 298 84 169 338 165 330 148 296 81 162 324 136 272 32 65 130 260 9 19 39 79 159 319 126 252 504 497 482 452 392 273 34 69 139 279 46 92 184 369 227 454 396 280 49 98 197 395 278 44 89 179 358 204 408 305 99 198 397 283 54 108 217 435 359 207 415 318 125 251 503 495 479 446 381 250 500 489 466 420 329 147 295 78 156 312 112 224 449 386 261 10 21 42 85 170 340 168 336 160 320 128 257 2 4 8 16 33 66 133 266 20 41 83 166 332 152 304 97 195 390 268 24 48 96 193 387 263 15 30 60 121 242 485 459 406 301 91 183 367 223 447 383 255 511 510 508 505 499 486 460 409 307 103 206 412 313 115 231 462 413 315 119 239 478 444 377 243 487 463 414 316 120 240 480 448 385 258 5 11 22 45 90 180 361 211 423 334 157 314 116 233 467 422 333 155 311 111 222 445 379 247 494 477 443 375 238 476 440 368 225 450 389 267 23 47 95 191 382 253 506 501 491 471 431 351 190 380 248 496 481 451 391 271 31 62 124 249 498 484 456 400 289 67 135 270 29 59 118 237 475 438 365 219 439 366 220 441 370 228 457 402 292 72 145 291 70 140 281 51 102 205 410 309 107 214 429 346 181 362 212 425 339 167 335 158 317 123 246 492 473 434 356 200 401 290 68 137 274 37 74 149 299 87 175 350 189 378 245 490 468 424 337 163 326 141 282 53 106 213 427 342 173 347 182 364 216 432 353 194 388 264 17 35 71 142 285 58 117 235 470 428 344 177 355 199 398 284 56 113 226 453 394 276 40 80 161 322 132 265 18 36 73 146 293 75 151 302 93 187 374 236 472 433 354 196 393 275 38 76 153 306 101 202 404 297

82 164 328 144 288 64 129 259 6 13 26 52 104 208 417 323 134 269 27 55 110 221 442 372 232 465 418 325 138 277 43 86 172 345 178 357 203 407 303 94 188 376 241 483 455 399 287 63 127 254 509 507 502 493 474 437 363 215 430 349 186 373 234 469 426 341 171 343 174 348 185 371 230 461 411 310 109 218 436 360 209 419 327 143 286 61 122 244 488 464 416 321 131 262 12 25 50 100 201 403 294 77 154 308 105 210 421 331 150 300 88 176 352 192 384 256

00000000010111011100111100010110111010110101000001110011001001111100001100101100000001100001011001000101111110010111001010111010011001111111001110001101000000111101111010010110100011001101001001001100000100101000100001111110110001110100001010110110100111001001011110110101100010101010111100100001001101111000000100010011101100111010100101010011110101110000011011100010010000010100001101010110011000110001000111110100010100110101111110001111001101101100101001000110110000111011111011101101111111110101010001110000 1

0 1 2 5 11 23 46 93 187 375 238 476 441 371 231 463 414 316 120 241 482 453 395 278 45 91 183 366 221 442 373 235 470 429 346 181 362 212 424 336 160 321 131 263 14 28 57 115 230 460 409 306 100 201 403 295 79 159 318 124 248 496 481 451 390 268 25 50 101 203 406 300 88 176 352 192 384 257 3 6 12 24 48 97 194 389 267 22 44 89 178 356 200 401 290 69 139 279 47 95 190 380 249 498 485 459 407 302 92 185 370 229 458 405 299 87 174 349 186 372 233 467 422 332 153 307 103 207 415 319 127 254 508 505 499 487 462 412 312 113 227 454 397 282 52 104 208 416 320 129 259 7 15 30 61 123 247 495 478 445 378 244 489 466 421 331 150 301 90 180 360 209 419 326 140 281 51 102 205 410 308 105 210 420 329 146 292 73 147 294 76 152 304 96 193 386 260 9 18 37 74 148 296 81 162 324 136 272 33 67 135 271 31 63 126 253 507 502 492 472 433 355 199 398 285 58 116 232 464 417 322 133 266 21 43 86 173 347 182 365 218 436 361 211 423 334 156 313 114 228 457 402 293 75 151 303 94 189 379 246 493 474 437 363 214 428 344 177 354 197 394 277 42 85 170 341 171 343 175 350 188 377 242 484 456 400 289 66 132 265 19 38 77 155 311 111 222 444 376 240 480 448 385 258 4 8 17 34 68 137 275 39 78 157 315 118 236 473 435 359 206 413 314 117 234 468 425 338 165 330 149 298 84 169 339 167 335 158 317 122 245 491 471 430 348 184 368 224 449 387 262 13 27 55 110 220 440 369 226 452 393 274 36 72 144 288 65 130 261 10 20 40 80 161 323 134 269 26 53 106 213 427 342 172 345 179 358 204 408 305 99 198 396 280 49 98 196 392 273 35 71 143 287 62 125 250 500 488 465 418 325 138 276 41 83 166 333 154 309 107 215 431 351 191 382 252 504 497 483 455 399 286 60 121 243 486 461 411 310 109 219 438 364 217 434 357 202 404 297 82 164 328 145 291 70 141 283 54 108 216 432 353 195 391 270 29 59 119 239 479 446 381 251 503 494 477 443 374 237 475 439 367 223 447 383 255 511 510 509 506 501 490 469 426 340 168 337 163 327 142 284 56 112 225 450 388 264 16 32 64 128 256

000000000110100110101011111010101011010010110101101111011100011011001111111001110100111110111100110010101001001111011001011000111111000010100011000110010011000101100100010101010000111111000100100100011010111011011011100101110100001000000101101100001101110101000100010011011111110110100011100000011100100101000001100111000101001010110011010000000100011110101111010001011110001110101100000111100100000000101011100001110110001000011000010011100111100000100101111111111010010000101110011011010100111011101111100101001100 11

0 1 3 6 13 26 52 105 211 422 333 154 309 106 213 427 343 175 351 190 381 250 501 490 469 426 341 171 342 173 346 180 361 210 421 331 150 301 90 181 363 214 429 347 183 367 222 445 379 247 494 476 440 369 227 454 397 283 54 108 217 435 359 207 415 319 127 254 508 505 499 487 462 413 314 116 233 467 423 335 159 318 125 251 503 495 478 444 377 243 486 460 409 306 101 202 405 298 84 169 338 164 329 147 295 79 158 317 123 246 492 473 434 357 203 406 300 88 177 355 199 399 287 63 126 252 504 496 481 450 389 266 20 40 81 163 326 140 280 49 99 198 396 281 50 100 201 403 294 76 152 305 98 197 395 278 44 89 178 356 200 401 290 69 138 277 42 85 170 340 168 336 161 323 135 271 31 62 124 248 497 482 452 393 274 36 73 146 292 72 145 291 70 141 282 53 107 215 430 349 187 374 237 475 438 365 219 439 366 220 441 370 229 459 407 302 93 186 372 232 464 417 322 132 264 16 32 65 130 261 11 22 45 91 182

364 216 432 353 195 390 269 27 55 110 221 442 373 234 468 424 337 162 324 136 273 34 68 137 275 38 77 155 311 111 223 447 382 253 507 502 493 474 436 360 209 419 327 142 284 56 112 224 448 385 259 7 14 28 57 114 228 457 402 293 74 148 296 80 160 321 131 262 12 25 51 103 206 412 312 113 226 453 394 276 41 82 165 330 149 299 86 172 345 179 358 205 410 308 104 208 416 320 129 258 4 8 17 35 71 143 286 61 122 245 491 471 431 350 189 378 244 488 465 418 325 139 279 47 94 188 376 241 483 455 398 285 58 117 235 470 428 344 176 352 193 387 263 15 30 60 121 242 484 456 400 288 64 128 257 2 5 10 21 43 87 174 348 184 368 225 451 391 270 29 59 118 236 472 433 354 196 392 272 33 67 134 268 24 48 97 194 388 265 19 39 78 156 313 115 231 463 414 316 120 240 480 449 386 260 9 18 37 75 151 303 95 191 383 255 511 510 509 506 500 489 466 420 328 144 289 66 133 267 23 46 92 185 371 230 461 411 310 109 218 437 362 212 425 339 167 334 157 315 119 238 477 443 375 239 479 446 380 249 498 485 458 404 297 83 166 332 153 307 102 204 408 304 96 192 384 256

00000000010110110110011100101010010111110000111000100011001010010011101101111001001001011010101000110100011101110000000100111100010100010101110101001100000100010110001011101111011000110001001101001110000100001001010110100001010101011000000011000010111100000111101000000111010110111010010100001100111111011010010001001000011111001011001011100011110011101000100000010100111110100110110101110011001000111111111010101111010111111001101010000011011000011010110011000111001111011101100100110011011111011111110001101110 01

0 1 2 5 11 22 45 91 182 365 219 438 364 217 435 359 206 412 313 114 229 458 405 298 84 169 338 165 331 151 303 95 190 380 248 496 481 451 391 270 28 56 113 226 452 392 273 35 70 140 281 50 101 202 404 297 82 164 329 147 295 78 157 315 118 237 475 439 367 222 444 377 242 484 457 402 292 73 146 293 75 150 301 90 181 362 213 426 340 168 337 163 326 141 282 52 104 209 419 327 142 285 59 119 238 476 440 368 224 448 385 258 4 9 19 39 79 158 316 120 241 482 453 394 276 40 81 162 325 138 277 43 87 174 349 186 373 234 468 425 339 166 332 152 304 96 193 386 260 8 17 34 69 139 278 44 88 177 354 197 395 279 46 93 187 375 239 478 445 379 246 492 472 433 355 198 396 280 49 98 196 393 275 38 77 154 308 105 211 423 334 156 312 112 225 450 388 264 16 33 66 132 265 18 37 74 149 299 86 173 346 180 360 208 417 322 133 266 21 42 85 170 341 171 342 172 344 176 352 192 384 257 3 6 12 24 48 97 194 389 267 23 47 94 188 376 240 480 449 387 263 15 30 61 122 244 488 464 416 320 129 259 7 14 29 58 117 235 470 429 347 183 366 221 442 372 233 466 421 330 148 296 80 161 323 134 268 25 51 103 207 415 319 126 253 507 502 493 474 436 361 210 420 328 145 290 68 137 274 36 72 144 289 67 135 271 31 62 124 249 498 485 459 406 300 89 178 357 203 407 302 92 184 369 227 455 399 286 60 121 243 487 462 413 314 116 232 465 418 324 136 272 32 65 130 261 10 20 41 83 167 335 159 318 125 250 500 489 467 422 333 155 310 109 218 437 363 215 430 348 185 371 230 460 409 306 100 200 401 291 71 143 287 63 127 255 511 510 509 506 501 490 469 427 343 175 350 189 378 245 491 471 431 351 191 382 252 505 499 486 461 410 309 106 212 424 336 160 321 131 262 13 27 54 108 216 432 353 195 390 269 26 53 107 214 428 345 179 358 204 408 305 99 199 398 284 57 115 231 463 414 317 123 247 494 477 443 374 236 473 434 356 201 403 294 76 153 307 102 205 411 311 111 223 446 381 251 503 495 479 447 383 254 508 504 497 483 454 397 283 55 110 220 441 370 228 456 400 288 64 128 256

00000000011111001010111010011110000111101001001101010011101000010001000000001010011000001110001100001100101100010111011000000110001001111101110101111010100010010010000011010000010110011010011011011010010110111101101110111100110001110101010010100001101111100000100001010110000010111100010001110011110111111010110110010100100010100011111100111000010010101010111111101100111011010101101011001000110011001111111110001111001000011101110001010100000010011001001011111010001011010001101100011010111001101110010011100101 11

0 1 3 7 15 31 62 124 249 498 485 458 405 299 87 174 349 186 372 233 467 423 335 158 316 120 240 481 451 391 271 30 61 122 244 489 466 420 329 147 294 77 154 309 106 212 425 339 167 334 157 314 116 232 464 417 322 132 264 17 34 68 136 272 32 64 128 257 2 5 10 20 41 83 166 332 152 304 96 193 387 263

14 28 56 113 227 454 396 280 48 97 195 390 268 25 50 101 203 406 300 88 177 354 197 395 279 46 93
187 374 236 472 432 352 192 385 259 6 12 24 49 98 196 393 275 39 79 159 318 125 251 503 494 477 442
373 235 471 431 350 189 378 245 490 468 424 337 162 324 137 274 36 73 146 292 72 144 288 65 131 262
13 26 52 104 208 416 321 130 261 11 22 44 89 179 358 205 410 308 105 211 422 333 155 310 109 219 438
365 218 436 361 210 421 331 150 301 91 183 367 222 445 379 246 493 475 439 366 221 443 375 239 478
444 377 243 486 460 408 305 99 199 398 285 58 117 234 469 426 340 169 338 165 330 148 296 80 161
323 134 269 27 55 111 223 446 380 248 496 480 449 386 260 8 16 33 66 133 266 21 43 86 172 344 176
353 194 389 267 23 47 94 188 376 241 482 452 392 273 35 71 142 284 57 115 231 463 414 317 123 247
495 479 447 382 253 506 501 491 470 429 347 182 364 217 434 357 202 404 297 82 164 328 145 290 69
138 276 40 81 163 327 143 287 63 126 252 505 499 487 462 412 312 112 225 450 388 265 18 37 74 149
298 85 170 341 171 343 175 351 191 383 254 509 507 502 492 473 435 359 206 413 315 118 237 474 437
362 213 427 342 173 346 181 363 214 428 345 178 356 200 401 291 70 140 281 51 102 204 409 307 103
207 415 319 127 255 511 510 508 504 497 483 455 399 286 60 121 242 484 456 400 289 67 135 270 29 59
119 238 476 440 369 226 453 394 277 42 84 168 336 160 320 129 258 4 9 19 38 76 153 306 100 201 402
293 75 151 303 95 190 381 250 500 488 465 418 325 139 278 45 90 180 360 209 419 326 141 283 54 108
216 433 355 198 397 282 53 107 215 430 348 185 371 230 461 411 311 110 220 441 370 228 457 403 295
78 156 313 114 229 459 407 302 92 184 368 224 448 384 256

000000000101010011001101000000011111110101001001100000100110101000111110100010100010010
010001011100111011000000110000111000101011001100100011010011111000011011111101111000100
011100100001001010111000010100101111100100101100011000111101011010101101110010101011110
0101101101011111111110001110100110111010101010000101101000110010100111010111011011001110
011011011110110010111010000111100111111001100010011110111011100011011000101100100111000
001101011000010001000001110111110110100101000001011110100100000001000011001111

0 1 2 5 10 21 42 84 169 339 166 332 153 307 102 205 410 308 104 208 416 320 128 257 3 7 15 31 63 127
254 509 506 501 490 468 425 338 164 329 147 294 76 152 304 96 193 386 260 9 19 38 77 154 309 106 212
424 337 163 327 143 287 62 125 250 500 488 465 418 325 138 276 40 81 162 324 137 274 36 73 146 292
72 145 290 69 139 279 46 92 185 371 231 462 413 315 118 236 472 432 352 192 385 259 6 12 24 48 97
195 391 270 28 56 113 226 453 394 277 43 86 172 345 179 358 204 409 306 100 200 401 291 70 141 282
52 105 211 423 335 159 318 124 248 496 481 451 390 269 27 55 111 223 447 382 253 507 503 495 478
444 376 241 482 452 392 273 35 71 142 284 57 114 228 456 400 289 66 132 265 18 37 74 149 299 87 174
348 184 368 225 450 389 266 20 41 82 165 331 151 303 95 190 380 249 498 484 457 402 293 75 150 300
88 177 355 198 396 280 49 99 199 399 286 61 122 245 491 470 429 346 181 362 213 427 342 173 347 183
366 220 441 370 229 458 405 298 85 171 343 175 350 188 377 242 485 459 406 301 91 182 365 218 437
363 215 431 351 191 383 255 511 510 508 504 497 483 455 398 285 58 116 233 467 422 333 155 311 110
221 442 373 234 469 426 341 170 340 168 336 161 322 133 267 22 45 90 180 360 209 419 326 140 281 50
101 202 404 297 83 167 334 157 314 117 235 471 430 349 187 374 237 475 438 364 217 435 359 206 412
313 115 230 461 411 310 109 219 439 367 222 445 379 246 492 473 434 357 203 407 302 93 186 372 232
464 417 323 135 271 30 60 121 243 487 463 415 319 126 252 505 499 486 460 408 305 98 196 393 275 39
79 158 317 123 247 494 477 443 375 238 476 440 369 227 454 397 283 54 108 216 433 354 197 395 278
44 89 178 356 201 403 295 78 156 312 112 224 449 387 262 13 26 53 107 214 428 344 176 353 194 388
264 17 34 68 136 272 32 65 131 263 14 29 59 119 239 479 446 381 251 502 493 474 436 361 210 421 330
148 296 80 160 321 130 261 11 23 47 94 189 378 244 489 466 420 328 144 288 64 129 258 4 8 16 33 67
134 268 25 51 103 207 414 316 120 240 480 448 384 256

000000000101000111000011100110010001010111111010101001100010001101100011010100010110100
100100111101100110000100010000110011110011111010010100111111000111110010010101100010111
101111010000010011100010011001101011001010100000000110000000100000110111011000000101110100
010010001111000101001000000111111100110111100101100001101001101000110001110100111011010

0001011001110101001011111000010101010111011100011001011100000111011111011011100101000001
11101011011001001101101101010110101110011100100001001011011111111101110101111

0 1 2 5 10 20 40 81 163 327 142 284 56 112 225 451 391 270 28 57 115 230 460 409 306 100 200 401 290 69 138 277 43 87 175 351 191 382 253 506 501 490 469 426 340 169 339 166 332 152 305 98 196 392 273 35 70 141 283 54 108 216 433 355 198 397 282 53 106 212 424 337 162 325 139 278 45 90 180 361 210 420 329 146 292 73 147 295 79 158 317 123 246 492 473 435 358 204 408 304 97 194 388 264 17 34 68 136 272 33 67 134 268 25 51 103 207 414 316 121 243 487 463 415 318 125 250 500 489 466 421 330 148 297 83 167 335 159 319 126 252 504 497 483 455 399 287 62 124 249 498 484 457 402 293 74 149 299 86 172 344 177 354 197 395 279 47 94 189 379 247 495 478 445 378 244 488 464 416 321 130 260 9 19 39 78 156 312 113 226 452 393 275 38 76 153 307 102 205 410 309 107 214 428 345 178 357 202 405 298 84 168 336 160 320 128 257 3 6 12 24 48 96 192 385 258 4 8 16 32 65 131 262 13 27 55 110 221 443 374 236 472 432 352 193 386 261 11 23 46 93 186 372 232 465 418 324 137 274 36 72 145 291 71 143 286 60 120 241 482 453 394 276 41 82 164 328 144 288 64 129 259 7 15 31 63 127 254 508 505 499 486 461 411 311 111 222 444 377 242 485 459 406 300 88 176 353 195 390 269 26 52 105 211 422 333 154 308 104 209 419 326 140 280 49 99 199 398 285 58 116 233 467 423 334 157 315 118 237 474 436 360 208 417 322 133 267 22 44 89 179 359 206 413 314 117 234 468 425 338 165 331 151 303 95 190 380 248 496 481 450 389 266 21 42 85 170 341 171 343 174 349 187 375 238 476 440 369 227 454 396 281 50 101 203 407 302 92 184 368 224 449 387 263 14 29 59 119 239 479 446 381 251 502 493 475 439 366 220 441 370 229 458 404 296 80 161 323 135 271 30 61 122 245 491 470 429 347 182 364 217 434 356 201 403 294 77 155 310 109 219 438 365 218 437 362 213 427 342 173 346 181 363 215 430 348 185 371 231 462 412 313 114 228 456 400 289 66 132 265 18 37 75 150 301 91 183 367 223 447 383 255 511 510 509 507 503 494 477 442 373 235 471 431 350 188 376 240 480 448 384 256

Appendix B

Copy of the Excel sheet showing the first cross join pair operation for n=5,6,7,8,9.

B.1. Cross-Join pair operation for n-5.

	DISTANCE	CONJUC.	ISA	DBS			NISA	NDBS		Verif.		30
	31	0	0	0			0	0		0		
	12	1	1	0			1	0		0		
	7	3	3	0			3	0		0		
	16	7	7	0			7	0		0		
	1	15	15	0			15	0		0		
	1	15	31	1			31	1		0		
	14	14	30	1			30	1		0		
	5	12	28	1			28	1		0		
	20	9	25	1			25	1		0		
	7	3	19	1			19	1		0		
	13	6	6	0			6	0		0		
	5	12	12	0			12	0		0		
	18	8	24	1			24	1		0		
	12	1	17	1			17	1		0		
	14	2	2	0			2	0		0		
	2	5	5	0			5	0		0		
	9	10	10	0			10	0		0		
	2	5	21	1			21	1		0		
	4	11	11	0			11	0		0		
	16	7	23	1			22	1		0		
	14	14	14	0			13	0		0		
	3	13	29	1			27	1		0		
	4	11	27	1			23	1		0		
	13	6	22	1			14	0		-1		
	3	13	13	0			29	1		1		
	9	10	26	1			26	1		0		
	3	4	20	1			20	1		0		
	20	9	9	0			9	0		0		
	14	2	18	1			18	1		0		
	3	4	4	0			4	0		0		
	18	8	8	0			8	0		0		
	31	0	16	1			16	1		0		
										2		

B.2. Cross-Join pair operation for n-6.

	Distance	CONJUG.	ISA	DBS			NISA	NDBS	Ver.		60		
	1	0	0	0			0	0	0		0	0	0
	9	1	1	0			1	0	0		1	0	0
	33	2	2	0			2	0	0		2	0	0
	40	5	5	0			5	0	0		5	0	0
	2	10	10	0			10	0	0		10	0	0
	52	21	21	0			21	0	0		21	0	0
	2	10	42	1			42	1	0		42	1	0
	23	20	20	0			20	0	0		20	0	0
	25	8	40	1			40	1	0		40	1	0
	53	16	16	0			16	0	0		16	0	0
	9	1	33	1			33	1	0		33	1	0
	12	3	3	0			3	0	0		3	0	0
	27	6	6	0			6	0	0		6	0	0
	42	13	13	0			13	0	0		13	0	0
	39	27	27	0			27	0	0		27	0	0
	30	23	55	1			55	1	0		55	1	0
	9	15	47	1			47	1	0		47	1	0
	1	31	31	0			31	0	0		30	0	0
	1	31	63	1			63	1	0		60	1	0
	7	30	62	1			62	1	0		56	1	0
	27	28	60	1			60	1	0		49	1	0
	40	24	56	1			56	1	0		35	1	0
	12	17	49	1			49	1	0		7	0	-1
	12	3	35	1			35	1	0		15	0	-1
	26	7	7	0			7	0	0		31	0	0
	9	15	15	0			15	0	0		63	1	1
	7	30	30	0			30	0	0		62	1	1
	25	29	61	1			61	1	0		61	1	0
	28	26	58	1			58	1	0		58	1	0
	23	20	52	1			52	1	0		52	1	0
	7	9	41	1			41	1	0		41	1	0
	11	18	18	0			18	0	0		18	0	0
	4	4	36	1			36	1	0		36	1	0
	25	8	8	0			8	0	0		8	0	0
	12	17	17	0			17	0	0		17	0	0
	33	2	34	1			34	1	0		34	1	0
	4	4	4	0			4	0	0		4	0	0
	7	9	9	0			9	0	0		9	0	0
	11	19	19	0			19	0	0		19	0	0
	27	6	38	1			38	1	0		38	1	0
	20	12	12	0			12	0	0		12	0	0
	7	25	25	0			25	0	0		25	0	0
	11	18	50	1			51	1	0		50	1	0
	40	5	37	1			39	1	0		37	1	0
	14	11	11	0			14	0	0	0	11	0	0
	30	23	23	0			28	0	0		23	0	0
	5	14	46	1			57	1	0		46	1	0
	27	28	28	0			50	1	1		28	0	0
	7	25	57	1			37	1	0		57	1	0
	11	19	51	1			11	0	-1		51	1	0
	26	7	39	1			23	0	-1		39	1	0
	5	14	14	0			46	1	1		14	0	0
	25	29	29	0			29	0	0		29	0	0
	39	27	59	1			59	1	0		59	1	0
	5	22	54	1			54	1	0		54	1	0
	42	13	45	1			45	1	0		45	1	0
	28	26	26	0			26	0	0		26	0	0
	52	21	53	1			53	1	0		53	1	0
	14	11	43	1			43	1	0		43	1	0
	5	22	22	0			22	0	0		22	0	0
	20	12	44	1			44	1	0		44	1	0
	40	24	24	0			24	0	0		24	0	0
	53	16	48	1			48	1	0		48	1	0
	1	0	32	1			32	1	0		32	1	0
									4				4

B.3. Cross-Join pair operation for n-6.

A/A		DISTANCE	CONJUG>	ISA	DBS			NISA	NDBS	Ver.		124
1		1	0	0	0			0	0	0		
2		11	1	1	0			1	0	0		
3		94	2	2	0			2	0	0		
4		51	5	5	0			5	0	0		
5		97	11	11	0			11	0	0		
6		117	22	22	0			22	0	0		
7		105	45	45	0			45	0	0		
8		112	26	90	1			90	1	0		
9		77	52	52	0			52	0	0		
10		42	40	104	1			104	1	0		
11		84	16	80	1			80	1	0		
12		115	32	32	0			32	0	0		
13		11	1	65	1			65	1	0		
14		67	3	3	0			3	0	0		
15		57	6	6	0			6	0	0		
16		74	12	12	0			12	0	0		
17		108	24	24	0			24	0	0		
18		52	49	49	0			49	0	0		
19		35	34	98	1			98	1	0		
20		79	4	68	1			68	1	0		
21		73	8	8	0			8	0	0		
22		31	17	17	0			17	0	0		
23		48	35	35	0			35	0	0		
24		58	7	71	1			71	1	0		
25		13	15	15	0			15	0	0		
26		41	30	30	0			30	0	0		
27		82	61	61	0			61	0	0		
28		57	58	122	1			122	1	0		
29		92	53	117	1			117	1	0		
30		28	42	106	1			106	1	0		
31		20	20	84	1			84	1	0		
32		55	41	41	0			41	0	0		
33		67	18	82	1			82	1	0		
34		59	36	36	0			36	0	0		
35		64	9	73	1			73	1	0		
36		52	19	19	0			19	0	0		
37		7	39	39	0			39	0	0		
38		13	15	79	1			78	1	0		
39		66	31	31	0			28	0	0		
40		74	62	62	0			57	0	0		
41		27	60	124	1			115	1	0		
42		5	57	121	1			103	1	0		
43		74	51	115	1			79	1	0		
44		7	39	103	1			31	0	-1		
45		38	14	78	1			62	0	-1		
46		31	28	28	0			124	1	1		
47		5	57	57	0			121	1	1		
48		46	50	114	1			114	1	0		
49		52	37	101	1			101	1	0		
50		6	10	74	1			74	1	0		
51		20	20	20	0			20	0	0		
52		42	40	40	0			40	0	0		
53		31	17	81	1			81	1	0		
54		35	34	34	0			34	0	0		
55		51	5	69	1			69	1	0		
56		6	10	10	0			10	0	0		
57		2	21	21	0			21	0	0		
58		28	42	42	0			42	0	0		
59		2	21	85	1			85	1	0		
60		62	43	43	0			43	0	0		
61		42	23	87	1			87	1	0		
62		14	46	46	0			46	0	0		
63		21	29	93	1			93	1	0		
64		46	59	59	0			59	0	0		
65		10	55	119	1			119	1	0		
66		38	47	111	1			111	1	0		
67		41	30	94	1			94	1	0		
68		27	60	60	0			60	0	0		
69		9	56	120	1			120	1	0		
70		52	49	113	1			113	1	0		
71		48	35	99	1			99	1	0		
72		57	6	70	1			70	1	0		
73		56	13	13	0			13	0	0		
74		39	27	27	0			27	0	0		
75		10	55	55	0			55	0	0		
76		14	46	110	1			110	1	0		
77		31	28	92	1			92	1	0		
78		9	56	56	0			56	0	0		
79		47	48	112	1			112	1	0		
80		16	33	97	1			97	1	0		
81		67	3	67	1			67	1	0		
82		58	7	7	0			7	0	0		
83		38	14	14	0			14	0	0		
84		21	29	29	0			29	0	0		
85		57	58	58	0			58	0	0		
86		77	52	116	1			116	1	0		
87		55	41	105	1			105	1	0		
88		52	19	83	1			83	1	0		
89		29	38	38	0			38	0	0		
90		74	12	76	1			76	1	0		
91		25	25	25	0			25	0	0		
92		46	50	50	0			50	0	0		
93		59	36	100	1			100	1	0		
94		73	8	72	1			72	1	0		
95		84	16	16	0			16	0	0		
96		16	33	33	0			33	0	0		
97		94	2	66	1			66	1	0		
98		79	4	4	0			4	0	0		
99		64	9	9	0			9	0	0		
100		67	18	18	0			18	0	0		
101		52	37	37	0			37	0	0		
102		97	11	75	1			75	1	0		
103		42	23	23	0			23	0	0		
104		38	47	47	0			47	0	0		
105		66	31	95	1			95	1	0		
106		1	63	63	0			63	0	0		
107		1	63	127	1			127	1	0		
108		78	62	126	1			126	1	0		
109		82	61	125	1			125	1	0		
110		46	59	123	1			123	1	0		
111		3	54	118	1			118	1	0		
112		105	45	109	1			109	1	0		
113		39	27	91	1			91	1	0		
114		3	54	54	0			54	0	0		
115		9	44	108	1			108	1	0		
116		25	25	89	1			89	1	0		
117		74	51	51	0			51	0	0		
118		29	38	102	1			102	1	0		
119		56	13	77	1			77	1	0		
120		112	26	26	0			26	0	0		
121		92	53	53	0			53	0	0		
122		62	43	107	1			107	1	0		
123		117	22	86	1			86	1	0		
124		9	44	44	0			44	0	0		
125		108	24	88	1			88	1	0		
126		47	48	48	0			48	0	0		
127		115	32	96	1			96	1	0		
128		1	0	64	1			64	1	0		
										4		

B.4. Cross-Join pair operation for n-7.

A/A		DISTANCE	CONJUG>	ISA	DBS			NISA	NDBS	Ver.		4	
1		1	0	0	0			0	0	0			
2		11	1	1	0			1	0	0			
3		94	2	2	0			2	0	0			
4		51	5	5	0			5	0	0			
5		97	11	11	0			11	0	0			
6		117	22	22	0			22	0	0			
7		105	45	45	0			45	0	0			
8		112	26	90	1			90	1	0			
9		77	52	52	0			52	0	0			
10		42	40	104	1			104	1	0			
11		84	16	80	1			80	1	0			
12		115	32	32	0			32	0	0			
13		11	1	65	1			65	1	0			
14		67	3	3	0			3	0	0			
15		57	6	6	0			6	0	0			
16		74	12	12	0			12	0	0			
17		108	24	24	0			24	0	0			
18		52	49	49	0			49	0	0			
19		35	34	98	1			98	1	0			
20		79	4	68	1			68	1	0			
21		73	8	8	0			8	0	0			
22		31	17	17	0			17	0	0			
23		48	35	35	0			35	0	0			
24		58	7	71	1			71	1	0			
25		13	15	15	0			15	0	0			
26		41	30	30	0			30	0	0			
27		82	61	61	0			61	0	0			
28		57	58	122	1			122	1	0			
29		92	53	117	1			117	1	0			
30		28	42	106	1			106	1	0			
31		20	20	84	1			84	1	0			
32		55	41	41	0			41	0	0			
33		67	18	82	1			82	1	0			
34		59	36	36	0			36	0	0			
35		64	9	73	1			73	1	0			
36		52	19	19	0			19	0	0			
37		7	39	39	0			39	0	0			
38		13	15	79	1			78	1	0			
39		66	31	31	0			28	0	0			
40		74	62	62	0			57	0	0			
41		27	60	124	1			115	1	0			
42		5	57	121	1			103	1	0			
43		74	51	115	1			79	1	0			
44		7	39	103	1			31	0	-1			
45		38	14	78	1			62	0	-1			
46		31	28	28	0			124	1	1			
47		5	57	57	0			121	1	1			
48		46	50	114	1			114	1	0			
49		52	37	101	1			101	1	0			
50		6	10	74	1			74	1	0			
51		20	20	20	0			20	0	0			
52		42	40	40	0			40	0	0			
53		31	17	81	1			81	1	0			
54		35	34	34	0			34	0	0			
55		51	5	69	1			69	1	0			
56		6	10	10	0			10	0	0			
57		2	21	21	0			21	0	0			
58		28	42	42	0			42	0	0			
59		2	21	85	1			85	1	0			
60		62	43	43	0			43	0	0			
61		42	23	87	1			87	1	0			
62		14	46	46	0			46	0	0			
63		21	29	93	1			93	1	0			
64		46	59	59	0			59	0	0			
65		10	55	119	1			119	1	0			
66		38	47	111	1			111	1	0			
67		41	30	94	1			94	1	0			
68		27	60	60	0			60	0	0			
69		9	56	120	1			120	1	0			
70		52	49	113	1			113	1	0			
71		48	35	99	1			99	1	0			
72		57	6	70	1			70	1	0			
73		56	13	13	0			13	0	0			
74		39	27	27	0			27	0	0			
75		10	55	55	0			55	0	0			
76		14	46	110	1			110	1	0			
77		31	28	92	1			92	1	0			
78		9	56	56	0			56	0	0			
79		47	48	112	1			112	1	0			
80		16	33	97	1			97	1	0			
81		67	3	67	1			67	1	0			
82		58	7	7	0			7	0	0			
83		38	14	14	0			14	0	0			
84		21	29	29	0			29	0	0			
85		57	58	58	0			58	0	0			
86		77	52	116	1			116	1	0			
87		55	41	105	1			105	1	0			
88		52	19	83	1			83	1	0			
89		29	38	38	0			38	0	0			
90		74	12	76	1			76	1	0			
91		25	25	25	0			25	0	0			
92		46	50	50	0			50	0	0			
93		59	36	100	1			100	1	0			
94		73	8	72	1			72	1	0			
95		84	16	16	0			16	0	0			
96		16	33	33	0			33	0	0			
97		94	2	66	1			66	1	0			
98		79	4	4	0			4	0	0			
99		64	9	9	0			9	0	0			
100		67	18	18	0			18	0	0			
101		52	37	37	0			37	0	0			
102		97	11	75	1			75	1	0			
103		42	23	23	0			23	0	0			
104		38	47	47	0			47	0	0			
105		66	31	95	1			95	1	0			
106		1	63	63	0			63	0	0			
107		1	63	127	1			127	1	0			
108		78	62	126	1			126	1	0			
109		82	61	125	1			125	1	0			
110		46	59	123	1			123	1	0			
111		3	54	118	1			118	1	0			
112		105	45	109	1			109	1	0			
113		39	27	91	1			91	1	0			
114		3	54	54	0			54	0	0			
115		9	44	108	1			108	1	0			
116		25	25	89	1			89	1	0			
117		74	51	51	0			51	0	0			
118		29	38	102	1			102	1	0			
119		56	13	77	1			77	1	0			
120		112	26	26	0			26	0	0			
121		92	53	53	0			53	0	0			
122		62	43	107	1			107	1	0			
123		117	22	86	1			86	1	0			
124		9	44	44	0			44	0	0			
125		108	24	88	1			88	1	0			
126		47	48	48	0			48	0	0			
127		115	32	96	1			96	1	0			
128		1	0	64	1			64	1	0			
										4			

B.5. Cross-Join pair operation for n-8.

	DISTANCE	CONJUG.	ISA	DBS	NISA	NDBS	Ver	250			
1	255	94	0	0	0	0	0		0	0	0
2	19	1	1	0	1	0	0		1	0	0
3	50	2	2	0	2	0	0		2	0	0
4	178	5	5	0	5	0	0		5	0	0
5	131	10	10	0	10	0	0		10	0	0
6	59	20	20	0	20	0	0		20	0	0
7	158	40	40	0	40	0	0		40	0	0
8	35	81	81	0	81	0	0		81	0	0
9	201	35	163	1	163	1	0		163	1	0
10	83	70	70	0	70	0	0		70	0	0
11	159	12	141	1	141	1	0		141	1	0
12	216	26	26	0	26	0	0		26	0	0
13	69	52	52	0	52	0	0		52	0	0
14	14	105	105	0	105	0	0		105	0	0
15	98	82	210	1	211	1	0		210	1	0
16	207	36	164	1	167	1	0		164	1	0
17	200	72	72	0	78	0	0		72	0	0
18	186	16	144	1	156	1	0		144	1	0
19	66	32	32	0	56	0	0		32	0	0
20	235	64	64	0	113	0	0		64	0	0
21	19	1	129	1	227	1	0		129	1	0
22	65	3	3	0	199	1	1		3	0	0
23	101	7	7	0	143	1	1		7	0	0
24	188	14	14	0	30	0	0		14	0	0
25	74	29	29	0	61	0	0		29	0	0
26	135	58	58	0	122	0	0		58	0	0
27	14	116	116	0	244	1	1		116	0	0
28	14	105	233	1	233	1	0		233	1	0
29	38	83	211	1	210	1	0		211	1	0
30	27	39	167	1	164	1	0		167	1	0
31	67	78	78	0	72	0	0		78	0	0
32	181	28	156	1	144	1	0		156	1	0
33	219	56	56	0	32	0	0		56	0	0
34	99	113	113	0	64	0	0		113	0	0
35	57	99	227	1	129	1	0		227	1	0
36	175	71	199	1	3	0	-1		199	1	0
37	88	15	143	1	7	0	-1		143	1	0
38	21	30	30	0	14	0	0		30	0	0
39	156	61	61	0	29	0	0		61	0	0
40	197	122	122	0	58	0	0		122	0	0
41	14	116	244	1	116	0	-1		244	1	0
42	41	104	232	1	232	1	0		232	1	0
43	35	81	209	1	209	1	0		209	1	0
44	175	34	162	1	162	1	0		162	1	0
45	90	69	69	0	69	0	0		69	0	0
46	137	11	139	1	139	1	0		139	1	0
47	199	22	22	0	22	0	0		22	0	0
48	193	44	44	0	44	0	0		44	0	0
49	150	88	88	0	88	0	0		88	0	0
50	129	48	176	1	176	1	0		176	1	0
51	203	96	96	0	96	0	0		96	0	0
52	34	65	193	1	193	1	0		193	1	0
53	50	2	130	1	130	1	0		130	1	0
54	153	4	4	0	4	0	0		4	0	0
55	166	9	9	0	9	0	0		9	0	0
56	169	19	19	0	19	0	0		19	0	0
57	27	39	39	0	39	0	0		39	0	0
58	95	79	79	0	79	0	0		79	0	0
59	21	30	158	1	158	1	0		158	1	0
60	59	60	60	0	60	0	0		60	0	0
61	89	121	121	0	121	0	0		121	0	0
62	153	114	242	1	242	1	0		242	1	0
63	181	101	229	1	229	1	0		229	1	0
64	12	74	202	1	202	1	0		202	1	0
65	59	20	148	1	148	1	0		148	1	0
66	46	41	41	0	41	0	0		41	0	0
67	38	83	83	0	83	0	0		83	0	0
68	158	38	166	1	166	1	0		166	1	0
69	108	76	76	0	76	0	0		76	0	0
70	25	25	153	1	153	1	0		153	1	0
71	172	50	50	0	50	0	0		50	0	0
72	144	100	100	0	100	0	0		100	0	0
73	151	73	201	1	201	1	0		201	1	0
74	148	18	146	1	146	1	0		146	1	0
75	39	37	37	0	37	0	0		37	0	0
76	12	74	74	0	74	0	0		74	0	0
77	60	21	149	1	149	1	0		149	1	0
78	62	43	43	0	43	0	0		43	0	0
79	161	86	86	0	86	0	0		86	0	0
80	167	45	173	1	173	1	0		173	1	0
81	25	90	90	0	90	0	0		90	0	0
82	69	52	180	1	180	1	0		180	1	0
83	41	104	104	0	104	0	0		104	0	0
84	82	80	208	1	208	1	0		208	1	0
85	66	32	160	1	160	1	0		160	1	0
86	34	65	65	0	65	0	0		65	0	0
87	65	3	131	1	131	1	0		131	1	0
88	81	6	6	0	6	0	0		6	0	0
89	5	12	12	0	12	0	0		12	0	0
90	88	24	24	0	24	0	0		24	0	0
91	109	49	49	0	49	0	0		49	0	0
92	57	99	99	0	99	0	0		99	0	0
93	83	70	198	1	198	1	0		198	1	0
94	5	12	140	1	140	1	0		140	1	0
95	25	25	25	0	25	0	0		25	0	0
96	79	51	51	0	51	0	0		51	0	0
97	55	103	103	0	103	0	0		103	0	0
98	67	78	206	1	206	1	0		206	1	0
99	74	29	157	1	157	1	0		157	1	0
100	44	59	59	0	59	0	0		59	0	0
101	96	118	118	0	118	0	0		118	0	0
102	3	109	237	1	237	1	0		237	1	0
103	145	91	219	1	219	1	0		219	1	0
104	68	54	182	1	182	1	0		182	1	0
105	3	109	109	0	109	0	0		109	0	0
106	25	90	218	1	218	1	0		218	1	0
107	122	53	181	1	181	1	0		181	1	0
108	55	106	106	0	106	0	0		106	0	0
109	30	85	213	1	213	1	0		213	1	0
110	28	42	170	1	170	1	0		170	1	0
111	53	84	84	0	84	0	0		84	0	0
112	46	41	169	1	169	1	0		169	1	0
113	98	82	82	0	82	0	0		82	0	0
114	39	37	165	1	165	1	0		165	1	0
115	130	75	75	0	75	0	0		75	0	0
116	68	23	151	1	151	1	0		151	1	0
117	115	47	47	0	47	0	0		47	0	0
118	76	94	94	0	94	0	0		94	0	0
119	59	60	188	1	188	1	0		188	1	0
120	12	120	120	0	120	0	0		120	0	0
121	132	112	240	1	240	1	0		240	1	0
122	58	97	225	1	225	1	0		225	1	0
123	45	67	195	1	195	1	0		195	1	0
124	101	7	135	1	135	1	0		135	1	0
125	88	15	15	0	15	0	0		15	0	0
126	28	31	31	0	31	0	0		31	0	0
127	107	63	63	0	63	0	0		63	0	0
128	1	127	127	0	127	0	0		127	0	0
129	1	127	255	1	255	1	0		255	1	0
130	105	126	254	1	254	1	0		254	1	0
131	18	124	252	1	252	1	0		252	1	0
132	12	120	248	1	248	1	0		249	1	0
133	99	113	241	1	241	1	0		243	1	0
134	67	98	226	1	226	1	0		231	1	0
135	90	69	197	1	197	1	0		207	1	0
136	131	10	138	1	138	1	0		159	1	0
137	60	21	21	0	21	0	0		62	0	0
138	28	42	42	0	42	0	0		124	0	0
139	30	85	85	0	85	0	0		248	1	1
140	62	43	171	1	171	1	0		241	1	0
141	90	87	87	0	87	0	0		226	1	1
142	43	46	174	1	174	1	0		197	1	0
143	17	93	93	0	93	0	0		138	1	1
144	44	59	187	1	187	1	0		21	0	-1
145	13	119	119	0	119	0	0		42	0	0
146	47	111	239	1	239	1	0		85	0	-1
147	86	95	223	1	223	1	0		171	1	0
148	7	62	190	1	190	1	0		87	0	-1
149	18	124	124	0	124	0	0		174	1	1
150	89	121	249	1	249	1	0		93	0	-1
151	37	115	243	1	243	1	0		187	1	0
152	55	103	231	1	231	1	0		119	0	-1
153	95	79	207	1	207	1	0		239	1	0
154	28	31	159	1	159	1	0		223	1	0
155	7	62	62	0	62	0	0		190	1	1
156	80	125	125	0	125	0	0		125	0	0
157	39	123	251	1	251	1	0		251	1	0
158	13	119	247	1	247	1	0		247	1	0
159	91	110	238	1	238	1	0		238	1	0
160	17	93	221	1	221	1	0		221	1	0
161	135	58	186	1	186	1	0		186	1	0
162	76	117	117	0	117	0	0		117	0	0
163	55	106	234	1	234	1	0		234	1	0
164	53	84	212	1	212	1	0		212	1	0
165	158	40	168	1	168	1	0		168	1	0
166	82	80	80	0	80	0	0		80	0	0
167	38	33	161	1	161	1	0		161	1	0
168	45	67	67	0	67	0	0		67	0	0
169	81	6	134	1	134	1	0		134	1	0
170	159	13	13	0	13	0	0		13	0	0
171	20	27	27	0	27	0	0		27	0	0
172	68	54	54	0	54	0	0		54	0	0
173	25	108	108	0	108	0	0		108	0	0
174	68	89	217	1	217	1	0		217	1	0
175	79	51	179	1	179	1	0		179	1	0
176	13	102	102	0	102	0	0		102	0	0
177	108	76	204	1	204	1	0		204	1	0
178	88	24	152	1	152	1	0		152	1	0
179	129	48	48	0	48	0	0		48	0	0
180	58	97	97	0	97	0	0		97	0	0
181	25	66	194	1	194	1	0		194	1	0
182	178	5	133	1	133	1	0		133	1	0
183	137	11	11	0	11	0	0		11	0	0
184	68	23	23	0	23	0	0		23	0	0
185	43	46	46	0	46	0	0		46	0	0
186	65	92	92	0	92	0	0		92	0	0
187	27	57	185	1	185	1	0		185	1	0
188	37	115	115	0	115	0	0		115	0	0
189	13	102	230	1	230	1	0		230	1	0
190	37	77	205	1	205	1	0		205	1	0
191	20	27	155	1	155	1	0		155	1	0
192	57	55	55	0	55	0	0		55	0	0
193	47	111	111	0	111	0	0		111	0	0
194	76	94	222	1	222	1	0		222	1	0
195	156	61	189	1	189	1	0		189	1	0
196	39	123	123	0	123	0	0		123	0	0
197	96	118	246	1	246	1	0		246	1	0
198	25	108	236	1	236	1	0		236	1	0
199	150	88	216	1	216	1	0		216	1	0
200	109	49	177	1	177	1	0		177	1	0
201	67	98	98	0	98	0	0		98	0	0
202	18	68	196	1	196	1	0		196	1	0
203	5	8	136	1	136	1	0		136	1	0
204	186	16	16	0	16	0	0		16	0	0
205	38	33	33	0	33	0	0		33	0	0
206	25	66	66	0	66	0	0		66	0	0
207	153	4	132	1	132	1	0		132	1	0
208	5	8	8	0	8	0	0		8	0	0
209	9	17	17	0	17	0	0		17	0	0
210	201	35	35	0	35	0	0		35	0	0
211	175	71	71	0	71	0	0		71	0	0
212	188	14	142	1	142	1	0		142	1	0
213	181	28	28	0	28	0	0		28	0	0
214	27	57	57	0	57	0	0		57	0	0
215	153	114	114	0	114	0	0		114	0	0
216	144	100	228	1	228	1	0		228	1	0
217	200	72	200	1	200	1	0		200	1	0
218	9	17	145	1	145	1	0		145	1	0
219	175	34	34	0	34	0	0		34	0	0
220	18	68	68	0	68	0	0		68	0	0
221	166	9	137	1	137	1	0		137	1	0
222	148	18	18	0	18	0	0		18	0	0
223	207	36	36	0	36	0	0		36	0	0
224	151	73	73	0	73	0	0		73	0	0
225	169	19	147	1	147	1	0		147	1	0
226	158	38	38	0	38	0	0		38	0	0
227	37	77	77	0	77	0	0		77	0	0
228	216	26	154	1	154	1	0		154	1	0
229	122	53	53	0	53	0	0		53	0	0
230	9	107	107	0	107	0	0		107	0	0
231	90	87	215	1	215	1	0		215	1	0
232	115	47	175	1	175	1	0		175	1	0
233	86	95	95	0	95	0	0		95	0	0
234	107	63	191	1	191	1	0		191	1	0
235	105	126	126	0	126	0	0		126	0	0
236	80	125	253	1	253	1	0		253	1	0
237	197	122	250	1	250	1	0		250	1	0
238	76	117	245	1	245	1	0		245	1	0
239	9	107	235	1	235	1	0		235	1	0
240	161	86	214	1	214	1	0		214	1	0
241	193	44	172	1	172	1	0		172	1	0
242	68	89	89	0	89	0	0		89	0	0
243	172	50	178	1	178	1	0		178	1	0
244	181	101	101	0	101	0	0		101	0	0
245	130	75	203	1	203	1	0		203	1	0
246	199	22	150	1	150	1	0		150	1	0
247	167	45	45	0	45	0	0		45	0	0
248	145	91	91	0	91	0	0		91	0	0
249	57	55	183	1	183	1	0		183	1	0
250	91	110	110	0	110	0	0		110	0	0
251	65	92	220	1	220	1	0		220	1	0
252	219	56	184	1	184	1	0		184	1	0
253	132	112	112	0	112	0	0		112	0	0
254	203	96	224	1	224	1	0		224	1	0
255	235	64	192	1	192	1	0		192	1	0
256	255	0	128	1	128	1	0		128	1	0
							6				10

B.6. Cross-Join pair operation for n-9.

Appendix c

Excel sheet showing the conditioning of the De Bruijn sequence in the corresponding Boolean function for n=5,6,7,8,9.

C1: Excel sheet showing the first conditioning of the De Bruijn sequence in the corresponding Boolean function for n=5.

ISA	DBS	NXTBIT		X0	X1	X2	X3	X4									
0	0	1	00000	0	0	0	0	0	0	0	**1**	00000	0	0	0	0	0
1	0	1	00001	0	0	0	0	1	16	1	**0**	10000	1	0	0	0	0
3	0	1	00011	0	0	0	1	1	8	0	**0**	01000	0	1	0	0	0
7	0	1	00111	0	0	1	1	1	24	1	**1**	11000	1	1	0	0	0
15	0	1	01111	0	1	1	1	1	4	0	**0**	00100	0	0	1	0	0
31	1	0	11111	1	1	1	1	1	20	1	**1**	10100	1	0	1	0	0
30	1	0	11110	1	1	1	1	0	12	0	**0**	01100	0	1	1	0	0
28	1	1	11100	1	1	1	0	0	28	1	**1**	11100	1	1	1	0	0
25	1	1	11001	1	1	0	0	1	2	0	**1**	00010	0	0	0	1	0
19	1	0	10011	1	0	0	1	1	18	1	**0**	10010	1	0	0	1	0
6	0	0	00110	0	0	1	1	0	10	0	**1**	01010	0	1	0	1	0
12	0	0	01100	0	1	1	0	0	26	1	**0**	11010	1	1	0	1	0
24	1	1	11000	1	1	0	0	0	6	0	**0**	00110	0	0	1	1	0
17	1	0	10001	1	0	0	0	1	22	1	**1**	10110	1	0	1	1	0
2	0	1	00010	0	0	0	1	0	14	0	**1**	01110	0	1	1	1	0
5	0	0	00101	0	0	1	0	1	30	1	**0**	11110	1	1	1	1	0
10	0	1	01010	0	1	0	1	0	1	0	**1**	00001	0	0	0	0	1
21	1	1	10101	1	0	1	0	1	17	1	**0**	10001	1	0	0	0	1
11	0	1	01011	0	1	0	1	1	9	0	**0**	01001	0	1	0	0	1
23	1	0	10111	1	0	1	1	1	25	1	**1**	11001	1	1	0	0	1
14	0	1	01110	0	1	1	1	0	5	0	**0**	00101	0	0	1	0	1
29	1	1	11101	1	1	1	0	1	21	1	**1**	10101	1	0	1	0	1
27	1	0	11011	1	1	0	1	1	13	0	**0**	01101	0	1	1	0	1
22	1	1	10110	1	0	1	1	0	29	1	**1**	11101	1	1	1	0	1
13	0	0	01101	**0**	**1**	**1**	0	1	3	0	**1**	00011	0	0	0	1	1
26	1	0	11010	1	1	0	1	0	19	1	**0**	10011	1	0	0	1	1
20	1	1	10100	1	0	1	0	0	11	0	**1**	01011	0	1	0	1	1
9	0	0	01001	0	1	0	0	1	27	1	**0**	11011	1	1	0	1	1
18	1	0	10010	1	0	0	1	0	7	0	**1**	00111	0	0	1	1	1
4	0	0	00100	0	0	1	0	0	23	1	**0**	10111	1	0	1	1	1
8	0	0	01000	0	1	0	0	0	15	0	**1**	01111	0	1	1	1	1
16	1	0	10000	1	0	0	0	0	31	1	**0**	11111	1	1	1	1	1

C2: Excel sheet showing the first conditioning of the De Bruijn sequence in the corresponding Boolean function for n=6.

.

ISA	DBS	NXTBIT																	
0	0	1	000000	0	0	0	0	0	0	0	0	**1**	000000	0	0	0	0	0	0
1	0	0	000001	0	0	0	0	0	1	32	1	**0**	100000	1	0	0	0	0	0
2	0	1	000010	0	0	0	0	1	0	16	0	**1**	010000	0	1	0	0	0	0
5	0	0	000101	0	0	0	1	0	1	48	1	**0**	110000	1	1	0	0	0	0
10	0	1	001010	0	0	1	0	1	0	8	0	**1**	001000	0	0	1	0	0	0
21	0	0	010101	0	1	0	1	0	1	40	1	**0**	101000	1	0	1	0	0	0
42	1	0	101010	1	0	1	0	1	0	24	0	**0**	011000	0	1	1	0	0	0
20	0	0	010100	0	1	0	1	0	0	56	1	**1**	111000	1	1	1	0	0	0
40	1	0	101000	1	0	1	0	0	0	4	0	**1**	000100	0	0	0	1	0	0
16	0	1	010000	0	1	0	0	0	0	36	1	**0**	100100	1	0	0	1	0	0
33	1	1	100001	1	0	0	0	0	1	20	0	**0**	010100	0	1	0	1	0	0
3	0	0	000011	0	0	0	0	1	1	52	1	**1**	110100	1	1	0	1	0	0
6	0	1	000110	0	0	0	1	1	0	12	0	**1**	001100	0	0	1	1	0	0
13	0	1	001101	0	0	1	1	0	1	44	1	**0**	101100	1	0	1	1	0	0
27	0	1	011011	0	1	1	0	1	1	28	0	**1**	011100	0	1	1	1	0	0
55	1	1	110111	1	1	0	1	1	1	60	1	**0**	111100	1	1	1	1	0	0
47	1	1	101111	1	0	1	1	1	1	2	0	**1**	000010	0	0	0	0	1	0
31	0	1	011111	0	1	1	1	1	1	34	1	**0**	100010	1	0	0	0	1	0
63	1	0	111111	1	1	1	1	1	1	18	0	**0**	010010	0	1	0	0	1	0
62	1	0	111110	1	1	1	1	1	0	50	1	**1**	110010	1	1	0	0	1	0
60	1	0	111100	1	1	1	1	0	0	10	0	**1**	001010	0	0	1	0	1	0
56	1	1	111000	1	1	1	0	0	0	42	1	**0**	101010	1	0	1	0	1	0
49	1	1	110001	1	1	0	0	0	1	26	0	**1**	011010	0	1	1	0	1	0
35	1	1	100011	1	0	0	0	1	1	58	1	**0**	111010	1	1	1	0	1	0
7	0	1	000111	0	0	0	1	1	1	6	0	**1**	000110	0	0	0	1	1	0
15	0	0	001111	0	0	1	1	1	1	38	1	**0**	100110	1	0	0	1	1	0
30	0	1	011110	0	1	1	1	1	0	22	0	**0**	010110	0	1	0	1	1	0
61	1	0	111101	1	1	1	1	0	1	54	1	**1**	110110	1	1	0	1	1	0
58	1	0	111010	1	1	1	0	1	0	14	0	**1**	001110	0	0	1	1	1	0
52	1	1	110100	1	1	0	1	0	0	46	1	**0**	101110	1	0	1	1	1	0
41	1	0	101001	1	0	1	0	0	1	30	0	**1**	011110	0	1	1	1	1	0
18	0	0	010010	0	1	0	0	1	0	62	1	**0**	111110	1	1	1	1	1	0
36	1	0	100100	1	0	0	1	0	0	1	0	**0**	000001	0	0	0	0	0	1
8	0	1	001000	0	0	1	0	0	0	33	1	**1**	100001	1	0	0	0	0	1
17	0	0	010001	0	1	0	0	0	1	17	0	**0**	010001	0	1	0	0	0	1
34	1	0	100010	1	0	0	0	1	0	49	1	**1**	110001	1	1	0	0	0	1
4	0	1	000100	0	0	0	1	0	0	9	0	**1**	001001	0	0	1	0	0	1
9	0	1	001001	0	0	1	0	0	1	41	1	**0**	101001	1	0	1	0	0	1
19	0	0	010011	0	1	0	0	1	1	25	0	**0**	011001	0	1	1	0	0	1
38	1	0	100110	1	0	0	1	1	0	57	1	**1**	111001	1	1	1	0	0	1
12	0	1	001100	0	0	1	1	0	0	5	0	**0**	000101	0	0	0	1	0	1
25	0	0	011001	0	1	1	0	0	1	37	1	**1**	100101	1	0	0	1	0	1
50	1	1	110010	1	1	0	0	1	0	21	0	**0**	010101	0	1	0	1	0	1
37	1	1	100101	1	0	0	1	0	1	53	1	**1**	110101	1	1	0	1	0	1
11	0	1	001011	0	0	1	0	1	1	13	0	**1**	001101	0	0	1	1	0	1
23	0	0	010111	0	1	0	1	1	1	45	1	**0**	101101	1	0	1	1	0	1
46	1	0	101110	1	0	1	1	1	0	29	0	**1**	011101	0	1	1	1	0	1
28	0	1	011100	0	1	1	1	0	0	61	1	**0**	111101	1	1	1	1	0	1
57	1	1	111001	1	1	1	0	0	1	3	0	**0**	000011	0	0	0	0	1	1
51	1	1	110011	1	1	0	0	1	1	35	1	**1**	100011	1	0	0	0	1	1
39	1	0	100111	1	0	0	1	1	1	19	0	**0**	010011	0	1	0	0	1	1
14	0	1	001110	0	0	1	1	1	0	51	1	**1**	110011	1	1	0	0	1	1
29	0	1	011101	0	1	1	1	0	1	11	0	**1**	001011	0	0	1	0	1	1
59	1	0	111011	1	1	1	0	1	1	43	1	**0**	101011	1	0	1	0	1	1
54	1	1	110110	1	1	0	1	1	0	27	0	**1**	011011	0	1	1	0	1	1
45	1	0	101101	1	0	1	1	0	1	59	1	**0**	111011	1	1	1	0	1	1
26	0	1	011010	0	1	1	0	1	0	7	0	**1**	000111	0	0	0	1	1	1
53	1	1	110101	1	1	0	1	0	1	39	1	**0**	100111	1	0	0	1	1	1
43	1	0	101011	1	0	1	0	1	1	23	0	**0**	010111	0	1	0	1	1	1
22	0	0	010110	0	1	0	1	1	0	55	1	**1**	110111	1	1	0	1	1	1
44	1	0	101100	1	0	1	1	0	0	15	0	**0**	001111	0	0	1	1	1	1
24	0	0	011000	0	1	1	0	0	0	47	1	**1**	101111	1	0	1	1	1	1
48	1	0	110000	1	1	0	0	0	0	31	0	**1**	011111	0	1	1	1	1	1
32	1	0	100000	1	0	0	0	0	0	63	1	**0**	111111	1	1	1	1	1	1

C3: Excel sheet showing the first conditioning of the De Bruijn sequence in the corresponding Boolean function for n=7.

ISA	DBS	NXTBIT																				
0	0	1	0000000	0	0	0	0	0	0	0	0	0	1	0000000	0	0	0	0	0	0	0
1	0	0	0000001	0	0	0	0	0	0	1	64	1	0	1000000	1	0	0	0	0	0	0
2	0	1	0000010	0	0	0	0	0	1	0	32	0	1	0100000	0	1	0	0	0	0	0
5	0	1	0000101	0	0	0	0	1	0	1	96	1	0	1100000	1	1	0	0	0	0	0
11	0	0	0001011	0	0	0	1	0	1	1	16	0	1	0010000	0	0	1	0	0	0	0
22	0	1	0010110	0	0	1	0	1	1	0	80	1	0	1010000	1	0	1	0	0	0	0
45	0	0	0101101	0	1	0	1	1	0	1	48	0	0	0110000	0	1	1	0	0	0	0
90	1	0	1011010	1	0	1	1	0	1	0	112	1	1	1110000	1	1	1	0	0	0	0
52	0	0	0110100	0	1	1	0	1	0	0	8	0	1	0001000	0	0	0	1	0	0	0
104	1	0	1101000	1	1	0	1	0	0	0	72	1	0	1001000	1	0	0	1	0	0	0
80	1	0	1010000	1	0	1	0	0	0	0	40	0	1	0101000	0	1	0	1	0	0	0
32	0	1	0100000	0	1	0	0	0	0	0	104	1	0	1101000	1	1	0	1	0	0	0
65	1	1	1000001	1	0	0	0	0	0	1	24	0	1	0011000	0	0	1	1	0	0	0
3	0	0	0000011	0	0	0	0	0	1	1	88	1	0	1011000	1	0	1	1	0	0	0
6	0	0	0000110	0	0	0	0	1	1	0	56	0	0	0111000	0	1	1	1	0	0	0
12	0	0	0001100	0	0	0	1	1	0	0	120	1	1	1111000	1	1	1	1	0	0	0
24	0	1	0011000	0	0	1	1	0	0	0	4	0	1	0000100	0	0	0	0	1	0	0
49	0	0	0110001	0	1	1	0	0	0	1	68	1	0	1000100	1	0	0	0	1	0	0
98	1	0	1100010	1	1	0	0	0	1	0	36	0	1	0100100	0	1	0	0	1	0	0
68	1	0	1000100	1	0	0	0	1	0	0	100	1	0	1100100	1	1	0	0	1	0	0
8	0	1	0001000	0	0	0	1	0	0	0	20	0	0	0010100	0	0	1	0	1	0	0
17	0	1	0010001	0	0	1	0	0	0	1	84	1	1	1010100	1	0	1	0	1	0	0
35	0	1	0100011	0	1	0	0	0	1	1	52	0	0	0110100	0	1	1	0	1	0	0
71	1	1	1000111	1	0	0	0	1	1	1	116	1	1	1110100	1	1	1	0	1	0	0
15	0	0	0001111	0	0	0	1	1	1	1	12	0	0	0001100	0	0	0	1	1	0	0
30	0	1	0011110	0	0	1	1	1	1	0	76	1	1	1001100	1	0	0	1	1	0	0
61	0	0	0111101	0	1	1	1	1	0	1	44	0	0	0101100	0	1	0	1	1	0	0
122	1	1	1111010	1	1	1	1	0	1	0	108	1	1	1101100	1	1	0	1	1	0	0
117	1	0	1110101	1	1	1	0	1	0	1	28	0	1	0011100	0	0	1	1	1	0	0
106	1	0	1101010	1	1	0	1	0	1	0	92	1	0	1011100	1	0	1	1	1	0	0
84	1	1	1010100	1	0	1	0	1	0	0	60	0	0	0111100	0	1	1	1	1	0	0
41	0	0	0101001	0	1	0	1	0	0	1	124	1	1	1111100	1	1	1	1	1	0	0
82	1	0	1010010	1	0	1	0	0	1	0	2	0	1	0000010	0	0	0	0	0	1	0
36	0	1	0100100	0	1	0	0	1	0	0	66	1	0	1000010	1	0	0	0	0	1	0
73	1	1	1001001	1	0	0	1	0	0	1	34	0	1	0100010	0	1	0	0	0	1	0
19	0	1	0010011	0	0	1	0	0	1	1	98	1	0	1100010	1	1	0	0	0	1	0
39	0	1	0100111	0	1	0	0	1	1	1	18	0	1	0010010	0	0	1	0	0	1	0
79	1	1	1001111	1	0	0	1	1	1	1	82	1	0	1010010	1	0	1	0	0	1	0
31	0	0	0011111	0	0	1	1	1	1	1	50	0	0	0110010	0	1	1	0	0	1	0
62	0	0	0111110	0	1	1	1	1	1	0	114	1	1	1110010	1	1	1	0	0	1	0
124	1	1	1111100	1	1	1	1	1	0	0	10	0	1	0001010	0	0	0	1	0	1	0
121	1	1	1111001	1	1	1	1	0	0	1	74	1	0	1001010	1	0	0	1	0	1	0
115	1	1	1110011	1	1	1	0	0	1	1	42	0	1	0101010	0	1	0	1	0	1	0
103	1	0	1100111	1	1	0	0	1	1	1	106	1	0	1101010	1	1	0	1	0	1	0
78	1	0	1001110	1	0	0	1	1	1	0	26	0	1	0011010	0	0	1	1	0	1	0
28	0	1	0011100	0	0	1	1	1	0	0	90	1	0	1011010	1	0	1	1	0	1	0
57	0	0	0111001	0	1	1	1	0	0	1	58	0	0	0111010	0	1	1	1	0	1	0
114	1	1	1110010	1	1	1	0	0	1	0	122	1	1	1111010	1	1	1	1	0	1	0
101	1	0	1100101	1	1	0	0	1	0	1	6	0	0	0000110	0	0	0	0	1	1	0
74	1	0	1001010	1	0	0	1	0	1	0	70	1	1	1000110	1	0	0	0	1	1	0
20	0	0	0010100	0	0	1	0	1	0	0	38	0	0	0100110	0	1	0	0	1	1	0
40	0	1	0101000	0	1	0	1	0	0	0	102	1	1	1100110	1	1	0	0	1	1	0
81	1	0	1010001	1	0	1	0	0	0	1	22	0	1	0010110	0	0	1	0	1	1	0
34	0	1	0100010	0	1	0	0	0	1	0	86	1	0	1010110	1	0	1	0	1	1	0
69	1	0	1000101	1	0	0	0	1	0	1	54	0	0	0110110	0	1	1	0	1	1	0
10	0	1	0001010	0	0	0	1	0	1	0	118	1	1	1110110	1	1	1	0	1	1	0
21	0	0	0010101	0	0	1	0	1	0	1	14	0	1	0001110	0	0	0	1	1	1	0
42	0	1	0101010	0	1	0	1	0	1	0	78	1	0	1001110	1	0	0	1	1	1	0
85	1	1	1010101	1	0	1	0	1	0	1	46	0	1	0101110	0	1	0	1	1	1	0
43	0	1	0101011	0	1	0	1	0	1	1	110	1	0	1101110	1	1	0	1	1	1	0
87	1	0	1010111	1	0	1	0	1	1	1	30	0	1	0011110	0	0	1	1	1	1	0
46	0	1	0101110	0	1	0	1	1	1	0	94	1	0	1011110	1	0	1	1	1	1	0
93	1	1	1011101	1	0	1	1	1	0	1	62	0	0	0111110	0	1	1	1	1	1	0
59	0	1	0111011	0	1	1	1	0	1	1	126	1	1	1111110	1	1	1	1	1	1	0
119	1	1	1110111	1	1	1	0	1	1	1	1	0	0	0000001	0	0	0	0	0	0	1
111	1	0	1101111	1	1	0	1	1	1	1	65	1	1	1000001	1	0	0	0	0	0	1
94	1	0	1011110	1	0	1	1	1	1	0	33	0	0	0100001	0	1	0	0	0	0	1
60	0	0	0111100	0	1	1	1	1	0	0	97	1	1	1100001	1	1	0	0	0	0	1
120	1	1	1111000	1	1	1	1	0	0	0	17	0	1	0010001	0	0	1	0	0	0	1
113	1	1	1110001	1	1	1	0	0	0	1	81	1	0	1010001	1	0	1	0	0	0	1
99	1	0	1100011	1	1	0	0	0	1	1	49	0	0	0110001	0	1	1	0	0	0	1
70	1	1	1000110	1	0	0	0	1	1	0	113	1	1	1110001	1	1	1	0	0	0	1
13	0	1	0001101	0	0	0	1	1	0	1	9	0	0	0001001	0	0	0	1	0	0	1
27	0	1	0011011	0	0	1	1	0	1	1	73	1	1	1001001	1	0	0	1	0	0	1
55	0	0	0110111	0	1	1	0	1	1	1	41	0	0	0101001	0	1	0	1	0	0	1
110	1	0	1101110	1	1	0	1	1	1	0	105	1	1	1101001	1	1	0	1	0	0	1
92	1	0	1011100	1	0	1	1	1	0	0	25	0	0	0011001	0	0	1	1	0	0	1
56	0	0	0111000	0	1	1	1	0	0	0	89	1	1	1011001	1	0	1	1	0	0	1
112	1	1	1110000	1	1	1	0	0	0	0	57	0	0	0111001	0	1	1	1	0	0	1
97	1	1	1100001	1	1	0	0	0	0	1	121	1	1	1111001	1	1	1	1	0	0	1
67	1	1	1000011	1	0	0	0	0	1	1	5	0	1	0000101	0	0	0	0	1	0	1
7	0	0	0000111	0	0	0	0	1	1	1	69	1	0	1000101	1	0	0	0	1	0	1
14	0	1	0001110	0	0	0	1	1	1	0	37	0	1	0100101	0	1	0	0	1	0	1
29	0	0	0011101	0	0	1	1	1	0	1	101	1	0	1100101	1	1	0	0	1	0	1
58	0	0	0111010	0	1	1	1	0	1	0	21	0	0	0010101	0	0	1	0	1	0	1
116	1	1	1110100	1	1	1	0	1	0	0	85	1	1	1010101	1	0	1	0	1	0	1
105	1	1	1101001	1	1	0	1	0	0	1	53	0	1	0110101	0	1	1	0	1	0	1
83	1	0	1010011	1	0	1	0	0	1	1	117	1	0	1110101	1	1	1	0	1	0	1
38	0	0	0100110	0	1	0	0	1	1	0	13	0	1	0001101	0	0	0	1	1	0	1
76	1	1	1001100	1	0	0	1	1	0	0	77	1	0	1001101	1	0	0	1	1	0	1
25	0	0	0011001	0	0	1	1	0	0	1	45	0	0	0101101	0	1	0	1	1	0	1
50	0	0	0110010	0	1	1	0	0	1	0	109	1	1	1101101	1	1	0	1	1	0	1
100	1	0	1100100	1	1	0	0	1	0	0	29	0	0	0011101	0	0	1	1	1	0	1
72	1	0	1001000	1	0	0	1	0	0	0	93	1	1	1011101	1	0	1	1	1	0	1
16	0	1	0010000	0	0	1	0	0	0	0	61	0	0	0111101	0	1	1	1	1	0	1
33	0	0	0100001	0	1	0	0	0	0	1	125	1	1	1111101	1	1	1	1	1	0	1
66	1	0	1000010	1	0	0	0	0	1	0	3	0	0	0000011	0	0	0	0	0	1	1
4	0	1	0000100	0	0	0	0	1	0	0	67	1	1	1000011	1	0	0	0	0	1	1
9	0	0	0001001	0	0	0	1	0	0	1	35	0	1	0100011	0	1	0	0	0	1	1
18	0	1	0010010	0	0	1	0	0	1	0	99	1	0	1100011	1	1	0	0	0	1	1
37	0	1	0100101	0	1	0	0	1	0	1	19	0	1	0010011	0	0	1	0	0	1	1
75	1	1	1001011	1	0	0	1	0	1	1	83	1	0	1010011	1	0	1	0	0	1	1
23	0	1	0010111	0	0	1	0	1	1	1	51	0	0	0110011	0	1	1	0	0	1	1
47	0	1	0101111	0	1	0	1	1	1	1	115	1	1	1110011	1	1	1	0	0	1	1
95	1	1	1011111	1	0	1	1	1	1	1	11	0	0	0001011	0	0	0	1	0	1	1
63	0	1	0111111	0	1	1	1	1	1	1	75	1	1	1001011	1	0	0	1	0	1	1
127	1	0	1111111	1	1	1	1	1	1	1	43	0	1	0101011	0	1	0	1	0	1	1
126	1	1	1111110	1	1	1	1	1	1	0	107	1	0	1101011	1	1	0	1	0	1	1
125	1	1	1111101	1	1	1	1	1	0	1	27	0	1	0011011	0	0	1	1	0	1	1
123	1	0	1111011	1	1	1	1	0	1	1	91	1	0	1011011	1	0	1	1	0	1	1
118	1	1	1110110	1	1	1	0	1	1	0	59	0	1	0111011	0	1	1	1	0	1	1
109	1	1	1101101	1	1	0	1	1	0	1	123	1	0	1111011	1	1	1	1	0	1	1
91	1	0	1011011	1	0	1	1	0	1	1	7	0	0	0000111	0	0	0	0	1	1	1
54	0	0	0110110	0	1	1	0	1	1	0	71	1	1	1000111	1	0	0	0	1	1	1
108	1	1	1101100	1	1	0	1	1	0	0	39	0	1	0100111	0	1	0	0	1	1	1
89	1	1	1011001	1	0	1	1	0	0	1	103	1	0	1100111	1	1	0	0	1	1	1
51	0	0	0110011	0	1	1	0	0	1	1	23	0	1	0010111	0	0	1	0	1	1	1
102	1	1	1100110	1	1	0	0	1	1	0	87	1	0	1010111	1	0	1	0	1	1	1
77	1	0	1001101	1	0	0	1	1	0	1	55	0	0	0110111	0	1	1	0	1	1	1
26	0	1	0011010	0	0	1	1	0	1	0	119	1	1	1110111	1	1	1	0	1	1	1
53	0	1	0110101	0	1	1	0	1	0	1	15	0	0	0001111	0	0	0	1	1	1	1
107	1	0	1101011	1	1	0	1	0	1	1	79	1	1	1001111	1	0	0	1	1	1	1
86	1	0	1010110	1	0	1	0	1	1	0	47	0	1	0101111	0	1	0	1	1	1	1
44	0	0	0101100	0	1	0	1	1	0	0	111	1	0	1101111	1	1	0	1	1	1	1
88	1	0	1011000	1	0	1	1	0	0	0	31	0	0	0011111	0	0	1	1	1	1	1
48	0	0	0110000	0	1	1	0	0	0	0	95	1	1	1011111	1	0	1	1	1	1	1
96	1	0	1100000	1	1	0	0	0	0	0	63	0	1	0111111	0	1	1	1	1	1	1
64	1	0	1000000	1	0	0	0	0	0	0	127	1	0	1111111	1	1	1	1	1	1	1

C4: Excel sheet showing the first conditioning of the De Bruijn sequence in the corresponding Boolean function for n=8.

ISA	DBS																						
0	0	1	00000000	0	0	0	0	0	0	0	0	0	0	1	00000000	0	0	0	0	0	0	0	0
1	0	0	00000001	0	0	0	0	0	0	0	1	128	1	0	10000000	1	0	0	0	0	0	0	0
2	0	1	00000010	0	0	0	0	0	0	1	0	64	0	1	01000000	0	1	0	0	0	0	0	0
5	0	0	00000101	0	0	0	0	0	1	0	1	192	1	0	11000000	1	1	0	0	0	0	0	0
10	0	0	00001010	0	0	0	0	1	0	1	0	32	0	0	00100000	0	0	1	0	0	0	0	0
20	0	0	00010100	0	0	0	1	0	1	0	0	160	1	1	10100000	1	0	1	0	0	0	0	0
40	0	1	00101000	0	0	1	0	1	0	0	0	96	0	1	01100000	0	1	1	0	0	0	0	0
81	0	1	01010001	0	1	0	1	0	0	0	1	224	1	0	11100000	1	1	1	0	0	0	0	0
163	1	0	10100011	1	0	1	0	0	0	1	1	16	0	1	00010000	0	0	0	1	0	0	0	0
70	0	1	01000110	0	1	0	0	0	1	1	0	144	1	0	10010000	1	0	0	1	0	0	0	0
141	1	0	10001101	1	0	0	0	1	1	0	1	80	0	1	01010000	0	1	0	1	0	0	0	0
26	0	0	00011010	0	0	0	1	1	0	1	0	208	1	0	11010000	1	1	0	1	0	0	0	0
52	0	1	00110100	0	0	1	1	0	1	0	0	48	0	1	00110000	0	0	1	1	0	0	0	0
105	0	0	01101001	0	1	1	0	1	0	0	1	176	1	0	10110000	1	0	1	1	0	0	0	0
210	1	0	11010010	1	1	0	1	0	0	1	0	112	0	0	01110000	0	1	1	1	0	0	0	0
164	1	0	10100100	1	0	1	0	0	1	0	0	240	1	1	11110000	1	1	1	1	0	0	0	0
72	0	0	01001000	0	1	0	0	1	0	0	0	8	0	1	00001000	0	0	0	0	1	0	0	0
144	1	0	10010000	1	0	0	1	0	0	0	0	136	1	0	10001000	1	0	0	0	1	0	0	0
32	0	0	00100000	0	0	1	0	0	0	0	0	72	0	0	01001000	0	1	0	0	1	0	0	0
64	0	1	01000000	0	1	0	0	0	0	0	0	200	1	1	11001000	1	1	0	0	1	0	0	0
129	1	1	10000001	1	0	0	0	0	0	0	1	40	0	1	00101000	0	0	1	0	1	0	0	0
3	0	1	00000011	0	0	0	0	0	0	1	1	168	1	0	10101000	1	0	1	0	1	0	0	0
7	0	0	00000111	0	0	0	0	0	1	1	1	104	0	0	01101000	0	1	1	0	1	0	0	0
14	0	1	00001110	0	0	0	0	1	1	1	0	232	1	1	11101000	1	1	1	0	1	0	0	0
29	0	0	00011101	0	0	0	1	1	1	0	1	24	0	1	00011000	0	0	0	1	1	0	0	0
58	0	0	00111010	0	0	1	1	1	0	1	0	152	1	0	10011000	1	0	0	1	1	0	0	0
116	0	1	01110100	0	1	1	1	0	1	0	0	88	0	0	01011000	0	1	0	1	1	0	0	0
233	1	1	11101001	1	1	1	0	1	0	0	1	216	1	1	11011000	1	1	0	1	1	0	0	0
211	1	1	11010011	1	1	0	1	0	0	1	1	56	0	1	00111000	0	0	1	1	1	0	0	0
167	1	0	10100111	1	0	1	0	0	1	1	1	184	1	0	10111000	1	0	1	1	1	0	0	0
78	0	0	01001110	0	1	0	0	1	1	1	0	120	0	0	01111000	0	1	1	1	1	0	0	0
156	1	0	10011100	1	0	0	1	1	1	0	0	248	1	1	11111000	1	1	1	1	1	0	0	0
56	0	1	00111000	0	0	1	1	1	0	0	0	4	0	1	00000100	0	0	0	0	0	1	0	0
113	0	1	01110001	0	1	1	1	0	0	0	1	132	1	0	10000100	1	0	0	0	0	1	0	0
227	1	1	11100011	1	1	1	0	0	0	1	1	68	0	1	01000100	0	1	0	0	0	1	0	0
199	1	1	11000111	1	1	0	0	0	1	1	1	196	1	0	11000100	1	1	0	0	0	1	0	0
143	1	0	10001111	1	0	0	0	1	1	1	1	36	0	1	00100100	0	0	1	0	0	1	0	0
30	0	1	00011110	0	0	0	1	1	1	1	0	164	1	0	10100100	1	0	1	0	0	1	0	0
61	0	0	00111101	0	0	1	1	1	1	0	1	100	0	1	01100100	0	1	1	0	0	1	0	0
122	0	0	01111010	0	1	1	1	1	0	1	0	228	1	0	11100100	1	1	1	0	0	1	0	0
244	1	0	11110100	1	1	1	1	0	1	0	0	20	0	0	00010100	0	0	0	1	0	1	0	0
232	1	1	11101000	1	1	1	0	1	0	0	0	148	1	1	10010100	1	0	0	1	0	1	0	0
209	1	0	11010001	1	1	0	1	0	0	0	1	84	0	1	01010100	0	1	0	1	0	1	0	0
162	1	1	10100010	1	0	1	0	0	0	1	0	212	1	0	11010100	1	1	0	1	0	1	0	0
69	0	1	01000101	0	1	0	0	0	1	0	1	52	0	1	00110100	0	0	1	1	0	1	0	0
139	1	0	10001011	1	0	0	0	1	0	1	1	180	1	0	10110100	1	0	1	1	0	1	0	0
22	0	0	00010110	0	0	0	1	0	1	1	0	116	0	1	01110100	0	1	1	1	0	1	0	0
44	0	0	00101100	0	0	1	0	1	1	0	0	244	1	0	11110100	1	1	1	1	0	1	0	0
88	0	0	01011000	0	1	0	1	1	0	0	0	12	0	0	00001100	0	0	0	0	1	1	0	0
176	1	0	10110000	1	0	1	1	0	0	0	0	140	1	1	10001100	1	0	0	0	1	1	0	0
96	0	1	01100000	0	1	1	0	0	0	0	0	76	0	1	01001100	0	1	0	0	1	1	0	0
193	1	0	11000001	1	1	0	0	0	0	0	1	204	1	0	11001100	1	1	0	0	1	1	0	0
130	1	0	10000010	1	0	0	0	0	0	1	0	44	0	0	00101100	0	0	1	0	1	1	0	0
4	0	1	00000100	0	0	0	0	0	1	0	0	172	1	1	10101100	1	0	1	0	1	1	0	0
9	0	1	00001001	0	0	0	0	1	0	0	1	108	0	1	01101100	0	1	1	0	1	1	0	0
19	0	1	00010011	0	0	0	1	0	0	1	1	236	1	0	11101100	1	1	1	0	1	1	0	0
39	0	1	00100111	0	0	1	0	0	1	1	1	28	0	1	00011100	0	0	0	1	1	1	0	0
79	0	0	01001111	0	1	0	0	1	1	1	1	156	1	0	10011100	1	0	0	1	1	1	0	0
158	1	0	10011110	1	0	0	1	1	1	1	0	92	0	1	01011100	0	1	0	1	1	1	0	0
60	0	1	00111100	0	0	1	1	1	1	0	0	220	1	0	11011100	1	1	0	1	1	1	0	0
121	0	0	01111001	0	1	1	1	1	0	0	1	60	0	1	00111100	0	0	1	1	1	1	0	0
242	1	1	11110010	1	1	1	1	0	0	1	0	188	1	0	10111100	1	0	1	1	1	1	0	0
229	1	0	11100101	1	1	1	0	0	1	0	1	124	0	1	01111100	0	1	1	1	1	1	0	0
202	1	0	11001010	1	1	0	0	1	0	1	0	252	1	0	11111100	1	1	1	1	1	1	0	0
148	1	1	10010100	1	0	0	1	0	1	0	0	2	0	1	00000010	0	0	0	0	0	0	1	0
41	0	1	00101001	0	0	1	0	1	0	0	1	130	1	0	10000010	1	0	0	0	0	0	1	0
83	0	0	01010011	0	1	0	1	0	0	1	1	66	0	0	01000010	0	1	0	0	0	0	1	0
166	1	0	10100110	1	0	1	0	0	1	1	0	194	1	1	11000010	1	1	0	0	0	0	1	0
76	0	1	01001100	0	1	0	0	1	1	0	0	34	0	0	00100010	0	0	1	0	0	0	1	0
153	1	0	10011001	1	0	0	1	1	0	0	1	162	1	1	10100010	1	0	1	0	0	0	1	0
50	0	0	00110010	0	0	1	1	0	0	1	0	98	0	0	01100010	0	1	1	0	0	0	1	0
100	0	1	01100100	0	1	1	0	0	1	0	0	226	1	1	11100010	1	1	1	0	0	0	1	0
201	1	0	11001001	1	1	0	0	1	0	0	1	18	0	0	00010010	0	0	0	1	0	0	1	0
146	1	1	10010010	1	0	0	1	0	0	1	0	146	1	1	10010010	1	0	0	1	0	0	1	0
37	0	0	00100101	0	0	1	0	0	1	0	1	82	0	1	01010010	0	1	0	1	0	0	1	0
74	0	1	01001010	0	1	0	0	1	0	1	0	210	1	0	11010010	1	1	0	1	0	0	1	0
149	1	1	10010101	1	0	0	1	0	1	0	1	50	0	0	00110010	0	0	1	1	0	0	1	0
43	0	0	00101011	0	0	1	0	1	0	1	1	178	1	1	10110010	1	0	1	1	0	0	1	0
86	0	1	01010110	0	1	0	1	0	1	1	0	114	0	0	01110010	0	1	1	1	0	0	1	0
173	1	0	10101101	1	0	1	0	1	1	0	1	242	1	1	11110010	1	1	1	1	0	0	1	0
90	0	0	01011010	0	1	0	1	1	0	1	0	10	0	0	00001010	0	0	0	0	1	0	1	0
180	1	0	10110100	1	0	1	1	0	1	0	0	138	1	1	10001010	1	0	0	0	1	0	1	0
104	0	0	01101000	0	1	1	0	1	0	0	0	74	0	1	01001010	0	1	0	0	1	0	1	0
208	1	0	11010000	1	1	0	1	0	0	0	0	202	1	0	11001010	1	1	0	0	1	0	1	0
160	1	1	10100000	1	0	1	0	0	0	0	0	42	0	1	00101010	0	0	1	0	1	0	1	0
65	0	1	01000001	0	1	0	0	0	0	0	1	170	1	0	10101010	1	0	1	0	1	0	1	0
131	1	0	10000011	1	0	0	0	0	0	1	1	106	0	1	01101010	0	1	1	0	1	0	1	0
6	0	0	00000110	0	0	0	0	0	1	1	0	234	1	0	11101010	1	1	1	0	1	0	1	0
12	0	0	00001100	0	0	0	0	1	1	0	0	26	0	0	00011010	0	0	0	1	1	0	1	0
24	0	1	00011000	0	0	0	1	1	0	0	0	154	1	1	10011010	1	0	0	1	1	0	1	0
49	0	1	00110001	0	0	1	1	0	0	0	1	90	0	0	01011010	0	1	0	1	1	0	1	0
99	0	0	01100011	0	1	1	0	0	0	1	1	218	1	1	11011010	1	1	0	1	1	0	1	0
198	1	0	11000110	1	1	0	0	0	1	1	0	58	0	0	00111010	0	0	1	1	1	0	1	0
140	1	1	10001100	1	0	0	0	1	1	0	0	186	1	1	10111010	1	0	1	1	1	0	1	0
25	0	1	00011001	0	0	0	1	1	0	0	1	122	0	0	01111010	0	1	1	1	1	0	1	0
51	0	1	00110011	0	0	1	1	0	0	1	1	250	1	1	11111010	1	1	1	1	1	0	1	0
103	0	0	01100111	0	1	1	0	0	1	1	1	6	0	0	00000110	0	0	0	0	0	1	1	0
206	1	1	11001110	1	1	0	0	1	1	1	0	134	1	1	10000110	1	0	0	0	0	1	1	0
157	1	1	10011101	1	0	0	1	1	1	0	1	70	0	1	01000110	0	1	0	0	0	1	1	0
59	0	0	00111011	0	0	1	1	1	0	1	1	198	1	0	11000110	1	1	0	0	0	1	1	0
118	0	1	01110110	0	1	1	1	0	1	1	0	38	0	1	00100110	0	0	1	0	0	1	1	0
237	1	1	11101101	1	1	1	0	1	1	0	1	166	1	0	10100110	1	0	1	0	0	1	1	0
219	1	0	11011011	1	1	0	1	1	0	1	1	102	0	0	01100110	0	1	1	0	0	1	1	0
182	1	1	10110110	1	0	1	1	0	1	1	0	230	1	1	11100110	1	1	1	0	0	1	1	0
109	0	0	01101101	0	1	1	0	1	1	0	1	22	0	0	00010110	0	0	0	1	0	1	1	0
218	1	1	11011010	1	1	0	1	1	0	1	0	150	1	1	10010110	1	0	0	1	0	1	1	0
181	1	0	10110101	1	0	1	1	0	1	0	1	86	0	1	01010110	0	1	0	1	0	1	1	0
106	0	1	01101010	0	1	1	0	1	0	1	0	214	1	0	11010110	1	1	0	1	0	1	1	0
213	1	0	11010101	1	1	0	1	0	1	0	1	54	0	0	00110110	0	0	1	1	0	1	1	0
170	1	0	10101010	1	0	1	0	1	0	1	0	182	1	1	10110110	1	0	1	1	0	1	1	0
84	0	1	01010100	0	1	0	1	0	1	0	0	118	0	1	01110110	0	1	1	1	0	1	1	0
169	1	0	10101001	1	0	1	0	1	0	0	1	246	1	0	11110110	1	1	1	1	0	1	1	0
82	0	1	01010010	0	1	0	1	0	0	1	0	14	0	1	00001110	0	0	0	0	1	1	1	0
165	1	1	10100101	1	0	1	0	0	1	0	1	142	1	0	10001110	1	0	0	0	1	1	1	0
75	0	1	01001011	0	1	0	0	1	0	1	1	78	0	0	01001110	0	1	0	0	1	1	1	0
151	1	1	10010111	1	0	0	1	0	1	1	1	206	1	1	11001110	1	1	0	0	1	1	1	0
47	0	0	00101111	0	0	1	0	1	1	1	1	46	0	0	00101110	0	0	1	0	1	1	1	0
94	0	0	01011110	0	1	0	1	1	1	1	0	174	1	1	10101110	1	0	1	0	1	1	1	0
188	1	0	10111100	1	0	1	1	1	1	0	0	110	0	0	01101110	0	1	1	0	1	1	1	0
120	0	0	01111000	0	1	1	1	1	0	0	0	238	1	1	11101110	1	1	1	0	1	1	1	0
240	1	1	11110000	1	1	1	1	0	0	0	0	30	0	1	00011110	0	0	0	1	1	1	1	0
225	1	1	11100001	1	1	1	0	0	0	0	1	158	1	0	10011110	1	0	0	1	1	1	1	0
195	1	1	11000011	1	1	0	0	0	0	1	1	94	0	0	01011110	0	1	0	1	1	1	1	0
135	1	1	10000111	1	0	0	0	0	1	1	1	222	1	1	11011110	1	1	0	1	1	1	1	0
15	0	1	00001111	0	0	0	0	1	1	1	1	62	0	1	00111110	0	0	1	1	1	1	1	0
31	0	1	00011111	0	0	0	1	1	1	1	1	190	1	0	10111110	1	0	1	1	1	1	1	0
63	0	1	00111111	0	0	1	1	1	1	1	1	126	0	1	01111110	0	1	1	1	1	1	1	0
127	0	1	01111111	0	1	1	1	1	1	1	1	254	1	0	11111110	1	1	1	1	1	1	1	0
255	1	0	11111111	1	1	1	1	1	1	1	1	1	0	0	00000001	0	0	0	0	0	0	0	1
254	1	0	11111110	1	1	1	1	1	1	1	0	129	1	1	10000001	1	0	0	0	0	0	0	1
252	1	0	11111100	1	1	1	1	1	1	0	0	65	0	1	01000001	0	1	0	0	0	0	0	1
248	1	1	11111000	1	1	1	1	1	0	0	0	193	1	0	11000001	1	1	0	0	0	0	0	1
241	1	0	11110001	1	1	1	1	0	0	0	1	33	0	0	00100001	0	0	1	0	0	0	0	1
226	1	1	11100010	1	1	1	0	0	0	1	0	161	1	1	10100001	1	0	1	0	0	0	0	1
197	1	0	11000101	1	1	0	0	0	1	0	1	97	0	0	01100001	0	1	1	0	0	0	0	1
138	1	1	10001010	1	0	0	0	1	0	1	0	225	1	1	11100001	1	1	1	0	0	0	0	1
21	0	0	00010101	0	0	0	1	0	1	0	1	17	0	1	00010001	0	0	0	1	0	0	0	1
42	0	1	00101010	0	0	1	0	1	0	1	0	145	1	0	10010001	1	0	0	1	0	0	0	1
85	0	1	01010101	0	1	0	1	0	1	0	1	81	0	1	01010001	0	1	0	1	0	0	0	1
171	1	1	10101011	1	0	1	0	1	0	1	1	209	1	0	11010001	1	1	0	1	0	0	0	1
87	0	0	01010111	0	1	0	1	0	1	1	1	49	0	1	00110001	0	0	1	1	0	0	0	1
174	1	1	10101110	1	0	1	0	1	1	1	0	177	1	0	10110001	1	0	1	1	0	0	0	1
93	0	1	01011101	0	1	0	1	1	1	0	1	113	0	1	01110001	0	1	1	1	0	0	0	1
187	1	1	10111011	1	0	1	1	1	0	1	1	241	1	0	11110001	1	1	1	1	0	0	0	1
119	0	1	01110111	0	1	1	1	0	1	1	1	9	0	1	00001001	0	0	0	0	1	0	0	1
239	1	1	11101111	1	1	1	0	1	1	1	1	137	1	0	10001001	1	0	0	0	1	0	0	1
223	1	0	11011111	1	1	0	1	1	1	1	1	73	0	1	01001001	0	1	0	0	1	0	0	1
190	1	0	10111110	1	0	1	1	1	1	1	0	201	1	0	11001001	1	1	0	0	1	0	0	1
124	0	1	01111100	0	1	1	1	1	1	0	0	41	0	1	00101001	0	0	1	0	1	0	0	1
249	1	1	11111001	1	1	1	1	1	0	0	1	169	1	0	10101001	1	0	1	0	1	0	0	1
243	1	1	11110011	1	1	1	1	0	0	1	1	105	0	0	01101001	0	1	1	0	1	0	0	1
231	1	1	11100111	1	1	1	0	0	1	1	1	233	1	1	11101001	1	1	1	0	1	0	0	1
207	1	1	11001111	1	1	0	0	1	1	1	1	25	0	1	00011001	0	0	0	1	1	0	0	1
159	1	0	10011111	1	0	0	1	1	1	1	1	153	1	0	10011001	1	0	0	1	1	0	0	1
62	0	1	00111110	0	0	1	1	1	1	1	0	89	0	0	01011001	0	1	0	1	1	0	0	1
125	0	1	01111101	0	1	1	1	1	1	0	1	217	1	1	11011001	1	1	0	1	1	0	0	1
251	1	1	11111011	1	1	1	1	1	0	1	1	57	0	0	00111001	0	0	1	1	1	0	0	1
247	1	0	11110111	1	1	1	1	0	1	1	1	185	1	1	10111001	1	0	1	1	1	0	0	1
238	1	1	11101110	1	1	1	0	1	1	1	0	121	0	0	01111001	0	1	1	1	1	0	0	1
221	1	0	11011101	1	1	0	1	1	1	0	1	249	1	1	11111001	1	1	1	1	1	0	0	1
186	1	1	10111010	1	0	1	1	1	0	1	0	5	0	0	00000101	0	0	0	0	0	1	0	1
117	0	0	01110101	0	1	1	1	0	1	0	1	133	1	1	10000101	1	0	0	0	0	1	0	1
234	1	0	11101010	1	1	1	0	1	0	1	0	69	0	1	01000101	0	1	0	0	0	1	0	1
212	1	0	11010100	1	1	0	1	0	1	0	0	197	1	0	11000101	1	1	0	0	0	1	0	1
168	1	0	10101000	1	0	1	0	1	0	0	0	37	0	0	00100101	0	0	1	0	0	1	0	1
80	0	1	01010000	0	1	0	1	0	0	0	0	165	1	1	10100101	1	0	1	0	0	1	0	1
161	1	1	10100001	1	0	1	0	0	0	0	1	101	0	1	01100101	0	1	1	0	0	1	0	1
67	0	0	01000011	0	1	0	0	0	0	1	1	229	1	0	11100101	1	1	1	0	0	1	0	1
134	1	1	10000110	1	0	0	0	0	1	1	0	21	0	0	00010101	0	0	0	1	0	1	0	1
13	0	1	00001101	0	0	0	0	1	1	0	1	149	1	1	10010101	1	0	0	1	0	1	0	1
27	0	0	00011011	0	0	0	1	1	0	1	1	85	0	1	01010101	0	1	0	1	0	1	0	1
54	0	0	00110110	0	0	1	1	0	1	1	0	213	1	0	11010101	1	1	0	1	0	1	0	1
108	0	1	01101100	0	1	1	0	1	1	0	0	53	0	1	00110101	0	0	1	1	0	1	0	1
217	1	1	11011001	1	1	0	1	1	0	0	1	181	1	0	10110101	1	0	1	1	0	1	0	1
179	1	0	10110011	1	0	1	1	0	0	1	1	117	0	0	01110101	0	1	1	1	0	1	0	1
102	0	0	01100110	0	1	1	0	0	1	1	0	245	1	1	11110101	1	1	1	1	0	1	0	1
204	1	0	11001100	1	1	0	0	1	1	0	0	13	0	1	00001101	0	0	0	0	1	1	0	1
152	1	0	10011000	1	0	0	1	1	0	0	0	141	1	0	10001101	1	0	0	0	1	1	0	1
48	0	1	00110000	0	0	1	1	0	0	0	0	77	0	0	01001101	0	1	0	0	1	1	0	1
97	0	0	01100001	0	1	1	0	0	0	0	1	205	1	1	11001101	1	1	0	0	1	1	0	1
194	1	1	11000010	1	1	0	0	0	0	1	0	45	0	1	00101101	0	0	1	0	1	1	0	1
133	1	1	10000101	1	0	0	0	0	1	0	1	173	1	0	10101101	1	0	1	0	1	1	0	1
11	0	1	00001011	0	0	0	0	1	0	1	1	109	0	0	01101101	0	1	1	0	1	1	0	1
23	0	0	00010111	0	0	0	1	0	1	1	1	237	1	1	11101101	1	1	1	0	1	1	0	1
46	0	0	00101110	0	0	1	0	1	1	1	0	29	0	0	00011101	0	0	0	1	1	1	0	1
92	0	1	01011100	0	1	0	1	1	1	0	0	157	1	1	10011101	1	0	0	1	1	1	0	1
185	1	1	10111001	1	0	1	1	1	0	0	1	93	0	1	01011101	0	1	0	1	1	1	0	1
115	0	0	01110011	0	1	1	1	0	0	1	1	221	1	0	11011101	1	1	0	1	1	1	0	1
230	1	1	11100110	1	1	1	0	0	1	1	0	61	0	0	00111101	0	0	1	1	1	1	0	1
205	1	1	11001101	1	1	0	0	1	1	0	1	189	1	1	10111101	1	0	1	1	1	1	0	1
155	1	1	10011011	1	0	0	1	1	0	1	1	125	0	1	01111101	0	1	1	1	1	1	0	1
55	0	1	00110111	0	0	1	1	0	1	1	1	253	1	0	11111101	1	1	1	1	1	1	0	1
111	0	0	01101111	0	1	1	0	1	1	1	1	3	0	1	00000011	0	0	0	0	0	0	1	1
222	1	1	11011110	1	1	0	1	1	1	1	0	131	1	0	10000011	1	0	0	0	0	0	1	1
189	1	1	10111101	1	0	1	1	1	1	0	1	67	0	0	01000011	0	1	0	0	0	0	1	1
123	0	1	01111011	0	1	1	1	1	0	1	1	195	1	1	11000011	1	1	0	0	0	0	1	1
246	1	0	11110110	1	1	1	1	0	1	1	0	35	0	1	00100011	0	0	1	0	0	0	1	1
236	1	0	11101100	1	1	1	0	1	1	0	0	163	1	0	10100011	1	0	1	0	0	0	1	1
216	1	1	11011000	1	1	0	1	1	0	0	0	99	0	0	01100011	0	1	1	0	0	0	1	1
177	1	0	10110001	1	0	1	1	0	0	0	1	227	1	1	11100011	1	1	1	0	0	0	1	1
98	0	0	01100010	0	1	1	0	0	0	1	0	19	0	1	00010011	0	0	0	1	0	0	1	1
196	1	0	11000100	1	1	0	0	0	1	0	0	147	1	0	10010011	1	0	0	1	0	0	1	1
136	1	0	10001000	1	0	0	0	1	0	0	0	83	0	0	01010011	0	1	0	1	0	0	1	1
16	0	1	00010000	0	0	0	1	0	0	0	0	211	1	1	11010011	1	1	0	1	0	0	1	1
33	0	0	00100001	0	0	1	0	0	0	0	1	51	0	1	00110011	0	0	1	1	0	0	1	1
66	0	0	01000010	0	1	0	0	0	0	1	0	179	1	0	10110011	1	0	1	1	0	0	1	1
132	1	0	10000100	1	0	0	0	0	1	0	0	115	0	0	01110011	0	1	1	1	0	0	1	1
8	0	1	00001000	0	0	0	0	1	0	0	0	243	1	1	11110011	1	1	1	1	0	0	1	1
17	0	1	00010001	0	0	0	1	0	0	0	1	11	0	1	00001011	0	0	0	0	1	0	1	1
35	0	1	00100011	0	0	1	0	0	0	1	1	139	1	0	10001011	1	0	0	0	1	0	1	1
71	0	0	01000111	0	1	0	0	0	1	1	1	75	0	1	01001011	0	1	0	0	1	0	1	1
142	1	0	10001110	1	0	0	0	1	1	1	0	203	1	0	11001011	1	1	0	0	1	0	1	1
28	0	1	00011100	0	0	0	1	1	1	0	0	43	0	0	00101011	0	0	1	0	1	0	1	1
57	0	0	00111001	0	0	1	1	1	0	0	1	171	1	1	10101011	1	0	1	0	1	0	1	1
114	0	0	01110010	0	1	1	1	0	0	1	0	107	0	1	01101011	0	1	1	0	1	0	1	1
228	1	0	11100100	1	1	1	0	0	1	0	0	235	1	0	11101011	1	1	1	0	1	0	1	1
200	1	1	11001000	1	1	0	0	1	0	0	0	27	0	0	00011011	0	0	0	1	1	0	1	1
145	1	0	10010001	1	0	0	1	0	0	0	1	155	1	1	10011011	1	0	0	1	1	0	1	1
34	0	0	00100010	0	0	1	0	0	0	1	0	91	0	1	01011011	0	1	0	1	1	0	1	1
68	0	1	01000100	0	1	0	0	0	1	0	0	219	1	0	11011011	1	1	0	1	1	0	1	1
137	1	0	10001001	1	0	0	0	1	0	0	1	59	0	0	00111011	0	0	1	1	1	0	1	1
18	0	0	00010010	0	0	0	1	0	0	1	0	187	1	1	10111011	1	0	1	1	1	0	1	1
36	0	1	00100100	0	0	1	0	0	1	0	0	123	0	0	01111011	0	1	1	1	1	0	1	1
73	0	1	01001001	0	1	0	0	1	0	0	1	251	1	1	11111011	1	1	1	1	1	0	1	1
147	1	0	10010011	1	0	0	1	0	0	1	1	7	0	0	00000111	0	0	0	0	0	1	1	1
38	0	1	00100110	0	0	1	0	0	1	1	0	135	1	1	10000111	1	0	0	0	0	1	1	1
77	0	0	01001101	0	1	0	0	1	1	0	1	71	0	0	01000111	0	1	0	0	0	1	1	1
154	1	1	10011010	1	0	0	1	1	0	1	0	199	1	1	11000111	1	1	0	0	0	1	1	1
53	0	1	00110101	0	0	1	1	0	1	0	1	39	0	1	00100111	0	0	1	0	0	1	1	1
107	0	1	01101011	0	1	1	0	1	0	1	1	167	1	0	10100111	1	0	1	0	0	1	1	1
215	1	1	11010111	1	1	0	1	0	1	1	1	103	0	0	01100111	0	1	1	0	0	1	1	1
175	1	1	10101111	1	0	1	0	1	1	1	1	231	1	1	11100111	1	1	1	0	0	1	1	1
95	0	1	01011111	0	1	0	1	1	1	1	1	23	0	0	00010111	0	0	0	1	0	1	1	1
191	1	0	10111111	1	0	1	1	1	1	1	1	151	1	1	10010111	1	0	0	1	0	1	1	1
126	0	1	01111110	0	1	1	1	1	1	1	0	87	0	0	01010111	0	1	0	1	0	1	1	1
253	1	0	11111101	1	1	1	1	1	1	0	1	215	1	1	11010111	1	1	0	1	0	1	1	1
250	1	1	11111010	1	1	1	1	1	0	1	0	55	0	1	00110111	0	0	1	1	0	1	1	1
245	1	1	11110101	1	1	1	1	0	1	0	1	183	1	0	10110111	1	0	1	1	0	1	1	1
235	1	0	11101011	1	1	1	0	1	0	1	1	119	0	1	01110111	0	1	1	1	0	1	1	1
214	1	0	11010110	1	1	0	1	0	1	1	0	247	1	0	11110111	1	1	1	1	0	1	1	1
172	1	1	10101100	1	0	1	0	1	1	0	0	15	0	1	00001111	0	0	0	0	1	1	1	1
89	0	0	01011001	0	1	0	1	1	0	0	1	143	1	0	10001111	1	0	0	0	1	1	1	1
178	1	1	10110010	1	0	1	1	0	0	1	0	79	0	0	01001111	0	1	0	0	1	1	1	1
101	0	1	01100101	0	1	1	0	0	1	0	1	207	1	1	11001111	1	1	0	0	1	1	1	1
203	1	0	11001011	1	1	0	0	1	0	1	1	47	0	0	00101111	0	0	1	0	1	1	1	1
150	1	1	10010110	1	0	0	1	0	1	1	0	175	1	1	10101111	1	0	1	0	1	1	1	1
45	0	1	00101101	0	0	1	0	1	1	0	1	111	0	0	01101111	0	1	1	0	1	1	1	1
91	0	1	01011011	0	1	0	1	1	0	1	1	239	1	1	11101111	1	1	1	0	1	1	1	1
183	1	0	10110111	1	0	1	1	0	1	1	1	31	0	1	00011111	0	0	0	1	1	1	1	1
110	0	0	01101110	0	1	1	0	1	1	1	0	159	1	0	10011111	1	0	0	1	1	1	1	1
220	1	0	11011100	1	1	0	1	1	1	0	0	95	0	1	01011111	0	1	0	1	1	1	1	1
184	1	0	10111000	1	0	1	1	1	0	0	0	223	1	0	11011111	1	1	0	1	1	1	1	1
112	0	0	01110000	0	1	1	1	0	0	0	0	63	0	1	00111111	0	0	1	1	1	1	1	1
224	1	0	11100000	1	1	1	0	0	0	0	0	191	1	0	10111111	1	0	1	1	1	1	1	1
192	1	0	11000000	1	1	0	0	0	0	0	0	127	0	1	01111111	0	1	1	1	1	1	1	1
128	1	0	10000000	1	0	0	0	0	0	0	0	255	1	0	11111111	1	1	1	1	1	1	1	1

C5: Excel sheet showing the first conditioning of the De Bruijn sequence in the corresponding Boolean function for n=9.