5-1-2013

Polynomial and Query Complexity of Minterm-Cyclic Functions

Matthew J. Harding
Dartmouth College

Dartmouth College Computer Science Technical Report TR2013-740

Polynomial and Query Complexity of Minterm-Cyclic Functions

A Thesis
Submitted to the Faculty
in partial fulfillment of the requirements for the
degree of

Bachelor of Arts

by

Matthew Harding

Dartmouth College
Hanover, New Hampshire

May 2013

Examining Committee:

Amit Chakrabarti

Scot Drysdale

Prasad Jayanti

ABSTRACT

Boolean functions are at the heart of all computations, and all Boolean functions can be reduced to a sum of pattern-matching functions, called *minterm-cyclic functions*. In this thesis, we examine properties of polynomials representing minterm-cyclic Boolean functions. We use the term *saturated* to represent a polynomial with degree equal to input size n for all n; this indicates the intuitive notion that such functions are in some way *complex*, or difficult to compute.

We present three main results. Firstly, there exist an infinite number of monotone minterm-cyclic functions that are not saturated. Secondly, for a specific class of minterms called *self-avoiding* minterms, we prove that the associated pattern-matching functions are not saturated; specifically, they can only have non-zero degree-n coefficients for n a multiple of the size of the minterm. Thirdly, for self-avoiding minterms $\alpha \in \{0, 1, *\}^k$ that contain some '$*$', the degree-n coefficients are always zero. These results may have implications in the fields of algebraic cryptographic attacks or efficiency of error-correcting codes.

Acknowledgements

I would like to extend my gratitude to all those who supported me throughout my research for this thesis. I would especially like to thank my advisor Dr. Amit Chakrabarti, whose insights, instruction and inspiration allowed me to obtain the results I did. I would also like to thank Dr. Sergi Elizalde and Dr. Rosa Orellana for their conversations about this material, which led to a better understanding and more complete proofs.

I am very grateful to Dr. Scot Drysdale and Dr. Prasad Jayanti for serving on my committee and offering their feedback. I would also like to thank Tim Tregubov and the Dartmouth College Department of Computer Science for the use of, and assistance with, the research computing cluster. Finally, I would like to express my appreciation for the many people who helped me in the stages of editing my thesis and presentation: particularly, John Harding, Ryan Archer, Matt Boyas, and Rachel Rosenberg. I appreciate the warm support of all my friends and family.

Contents

Chapter 1

Introduction and Purpose

1.1 Boolean Functions and Polynomial Representations

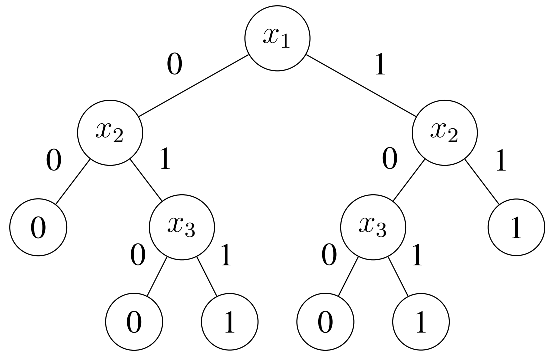
Boolean functions are some of the simplest and most fundamental areas of Computational Complexity theory. The study of Boolean functions can provide the building blocks to address larger, more complex questions, such as the tightening of the exponential bounds of NP-complete problems [2], and the efficiency of error-checking computations [3, 9].

A Boolean function is a function on a string of truth values that returns a single bit of output. Using '0' to represent 'FALSE' and '1' to represent 'TRUE', such a function takes the form $f : \{0,1\}^n \to \{0,1\}$. Throughout this thesis, we shall use n to denote the number of bits in the input to a Boolean function, and we shall represent the bits of input x as $x_1, x_2, \cdots, x_n$.

A binary decision tree is a deterministic method of querying bits of the input string to evaluate the output value. The decision tree consists of a set of vertices and directed edges, where each vertex has at most two children. (That is, each vertex is the source of at most two edges.) Each edge is labeled with either a '0' or a '1'. Starting at the 'root,' we descend the tree to compute the value of the function, querying one bit x_i at each vertex. The result of each query determines which labeled edge we follow. When we reach a leaf (a vertex with no children), the decision tree outputs a Boolean value. A decision tree *computes* f if the value returned by the decision tree on any input x is equal to the value $f(x)$. The height of such a tree is the maximum length (in edges) of a path from root to leaf.

Definition 1.1. The *query complexity*, or *deterministic decision tree complexity*, of a Boolean function f is the minimum height of a binary decision tree that computes f and is denoted $\mathrm{D}(f)$. If $\mathrm{D}(f)$ equals the size n of the input, then f is deemed *evasive*.

Example. Let $\textsc{Maj}_3$ be the 3-bit Majority function, which returns i if two or more bits of the input x equal i. Then the following decision tree represents $\textsc{Maj}_3$:



In this case, the tree's height is 3, and no smaller tree computes MAJ_3,[1] so $\text{D}(\text{MAJ}_3) = 3$.

Example. The AND_n function returns 1 if and only if all n bits of input are 1. The AND_n function must query all n bits in the worst case, where the input consists of only 1s; thus, AND_n is evasive with $\text{D}(\text{AND}_n) = n$.
(Note that we will sometimes drop the subscript n and write AND when the context is clear.)

We say that a polynomial $p_f : \mathbb{R}^n \to \mathbb{R}$ *represents* a Boolean function $f : \{0,1\}^n \to \{0,1\}$ if $p_f(x) = f(x) \; \forall \; x \in \{0,1\}^n$. A *multilinear* polynomial is defined as a polynomial p where no variable is raised to a power greater than 1. (We restrict ourselves to multilinear polynomials because for any $k > 1$ and $x_i \in \{0,1\}$, we have ${x_i}^k = x_i$.) The *degree* of such a polynomial is defined as the highest number of variables in a single product, when the polynomial is represented as a sum of products. Note that there is no ambiguity in referencing the coefficient of the degree-n term, because only one unique product can contain all n variables. (This is not the case for the degree-2 term, for instance, which could be x_1x_2 or x_1x_3, etc, depending on n.)

Example. The multilinear polynomial $p = x_1x_2x_3 + x_1x_2 + x_1$ has degree 3, and the multilinear polynomial $p' = x_1x_4 + x_2x_3 + 1$ has degree 2.

Theorem 1.2. *Every Boolean function f has a unique representation as a multilinear polynomial p_f.*

Proof. The intuition here is that we want to evaluate f at each of the 2^n possible binary inputs $\alpha \in \{0,1\}^n$ and use these values $f(\alpha)$ to construct p_f. To do so, we use the Kronecker delta $\delta_\alpha(x)$:

$$\delta_\alpha(x) = \begin{cases} 1 & \text{if } x = \alpha \\ 0 & \text{otherwise} \end{cases}$$

We represent $\delta_{\alpha_1\alpha_2\cdots\alpha_n}(x_1x_2\cdots x_n)$ as the polynomial $\big(\prod_{\substack{i\in[n]\\ \alpha_i=1}} x_i\big)\big(\prod_{\substack{i\in[n]\\ \alpha_i=0}}(1-x_i)\big)$, where $[n] = \{1, \cdots, n\}$. That way, if $x_i = 1$ and $\alpha_i = 0$ for any i, the product will evaluate to 0. Similarly, if $x_i = 0$ and $\alpha_i = 1$ for any i, the product will evaluate to 0. If, alternatively, $x_i = \alpha_i \; \forall \; 1 \le i \le n$, then the product will evaluate to 1. This exactly represents the desired behavior of δ_α.

[1] To see this, note that in the worst case the first two queries will be a 0 and a 1. This is inevitable by the design of a binary decision tree. Furthermore, in the general case of MAJ_n (which is defined only when n is odd), no decision tree computing MAJ_n can have height less than n. In the worst case, the first $n-1$ queries will result in $\frac{n-1}{2}$ 0s and $\frac{n-1}{2}$ 1s.

Then we interpolate the value of f at every point $\alpha \in \{0,1\}^n$ and multiply each Kronecker delta δ_α by the associated value $f(\alpha)$:

$$p_f(x) = \sum_{\alpha \in \{0,1\}^n} \delta_\alpha(x) f(\alpha) = \sum_{\substack{\alpha \in \{0,1\}^n \\ f(\alpha)=1}} \delta_\alpha(x) \tag{1.1}$$

The expansion of this sum gives us a polynomial $p_f : \mathbb{R}^n \to \mathbb{R}$. At any $x \in \{0,1\}$, $p_f(x) = \sum_{\alpha \in \{0,1\}^n} \delta_\alpha(x) f(\alpha) = f(x)$, so the polynomial p_f represents f.

Furthermore, note that each polynomial representing δ_α is multilinear by construction, so p_f is the sum of multilinear polynomials, which is multilinear.

Finally, we conclude with one possible proof of uniqueness. We note that any function $f : \{0,1\}^n \to \mathbb{R}$ can be interpolated in the fashion described above, so the 2^n Kronecker delta functions δ_α span the set of functions $\{f : \{0,1\}^n \to \mathbb{R}\}$. It should also be clear that they form a linearly independent set in the space $\{f : \{0,1\}^n \to \mathbb{R}\}$. (Take any linear combination f of elements δ_α that sums to the zero function. Then $f(\alpha) = 0 \,\forall\, \alpha \in \{0,1\}^n$, so all coefficients are zero.) This proves that the 2^n Kronecker delta functions δ_α form a basis for the set of functions $\{f : \{0,1\}^n \to \mathbb{R}\}$. In a finite-dimensional vector space such as $\{f : \{0,1\}^n \to \mathbb{R}\}$, it is known that every vector f has a unique expansion as a linear combination of basis elements. Thus, p_f is unique. □

Note that our uniqueness proof here resembles that of [8]. One can also prove uniqueness using Parseval's Theorem, as in [7].

Definition 1.3. The *polynomial complexity* of a Boolean function f is the degree of the unique multilinear polynomial representing f, and is denoted deg(f). For ease of notation, we will also sometimes reference 'the degree-n term (or coefficient) of f.' This is shorthand for 'the degree-n term (or coefficient) of the multilinear polynomial representing f.'

Definition 1.4. We define a Boolean function f as *saturated* if the multilinear polynomial representing f has degree n for every input size n.

Theorem 1.5. *deg*$(f) \leq D(f) \;\; \forall\, f : \{0,1\}^n \to \{0,1\}$

The following proof is influenced by [2]:

Proof. Given a Boolean function $f : \{0,1\}^n \to \{0,1\}$ and decision tree T of height $D(f)$ that computes f, we create a polynomial p in a similar manner to the proof of Theorem 1.2. For every leaf node t with output value 1, trace the path p from the root. Define

$$\delta_t = \begin{cases} 1 & \text{if we reach } t \in T \text{ on input } x \\ 0 & \text{otherwise} \end{cases}$$

Let $S_t \subseteq [n] = \{1, \cdots, n\}$ be the set of indices i s.t. x_i is queried along this path, and $x_i = 1$. Let T_t be the set of indices i s.t. x_i is queried along this path, and $x_i = 0$. Then represent δ_t by the polynomial $p_t := \prod_{i \in S_t} x_i \prod_{i \in T_t}(1 - x_i)$. p_t has degree at most $D(f)$, because the number of edges along this path is at most $D(f)$. Also, $p_t(x) = 1$ if and only if we reach $t \in T$ on input x.

We then sum these (multilinear) polynomials over all leaf nodes t with output value 1: $p := \sum_t p_t$. Whenever T on input x reaches a leaf node with output value 1, $p(x)$ evaluates to 1. Otherwise, $p(x)$ evaluates to 0. Since T computes f, then p also represents f. Furthermore, the degree of any individual p_t is at most $D(f)$, so the degree of their sum p is at most $D(f)$. That is, deg$(f) \leq D(f)$. □

Example. Let MAJ_3 be the 3-bit Majority function, as defined previously. Then the polynomial representing MAJ_3 is $p(x) = x_1x_2 + x_2x_3 + x_1x_3 - 2x_1x_2x_3$, so deg$(\text{MAJ}_3) = 3$. In fact, deg$(\text{MAJ}_n) = n$ for all (odd) n, so MAJ_n is saturated. (We will not prove this general result.) Also recall that $\text{D}(\text{MAJ}_3) = 3$, so for this example, deg(f) = D(f).

1.2 Minterms and Shifts

The following definitions lead to our concept of a *minterm*. We borrow some of the following terms from Chakraborty and others; see [4] for a complete look at their context within the realm of sensitivity and block sensitivity.

Say we have some n variables $x_1, \cdots, x_n$ that can each be given a value 0 or 1. Then we can choose some subset $S \subseteq [n] = \{1, \cdots, n\}$ and assign values to the variables x_i s.t. $i \in S$. This is called a *partial assignment*.

Definition 1.6. Formally, we define a *partial assignment* $p : S \to \{0, 1\}$, where $S \subseteq [n]$.

Definition 1.7. Given a function $f : \{0, 1\}^n \to \{0, 1\}$, we define a *1-certificate* as a partial assignment s.t. $f(x) = 1$ for every x that satisfies this partial assignment.

Definition 1.8. Finally, a *minterm* $p : S \to \{0, 1\}$ is defined as a 'minimal 1-certificate.' That is, p is a 1-certificate and if we restrict the domain S, we no longer have a 1-certificate.

The concept of a minterm is useful, but this string of definitions is unnecessarily complicated for our use. We find it more convenient to use an alternate view of minterms. We start with a pattern α of n symbols, consisting of 0s, 1s and *s. We define $S \subseteq [n]$ to be the symbols that are not *s. Then we define $p : S \to \{0, 1\}$ as the partial assignment that assigns the value 0 to each symbol '0' and the value 1 to each symbol '1.' We then consider the Kronecker delta function $\delta_p : \{0, 1\}^n \to \{0, 1\}$, where $\delta_p(x) = 1$ if and only if for all $i \in S$, $x_i = p(i)$. Notice that for every x that satisfies this partial assignment, we have $\delta_p(x) = 1$; thus, p is a 1-certificate w.r.t. δ_p. Finally, if we were to remove any index i from S, we could construct an input x' with bit x_i flipped; based on our construction of δ_p, we would have $\delta_p(x') = 0$, so we would no longer have a 1-certificate for δ_p. This shows that such a p is a minimal 1-certificate, or a *minterm*.

From this point, we rely on our definition of a pattern α as a minterm, and the previous definitions are implicit.

Definition 1.9. A *minterm* is a pattern of 0s, 1s, and *s, denoted $\alpha \in \{0, 1, *\}^k$. For convenience, we will interchangeably use subset notation to represent a minterm according to the indices of 0s (A_0) and 1s (A_1) : $\alpha = (A_0, A_1)$ where $A_0, A_1 \subseteq [k]$. Note that $[k]$ represents the set $[k] = \{1, 2, \cdots, n\}$. We will also use index notation $\alpha[i] \in \{0, 1, *\}$ to denote the ith symbol of α. Finally, we will use superscript notation c^k to represent the pattern of a symbol c repeated k times.

Example. We represent α = 10*1 as $(A_0 = \{2\}, A_1 = \{1, 4\})$. For this minterm, we have $\alpha[1] = 1, \alpha[2] = 0, \alpha[3] = *, \alpha[4] = 1$.

Example. We represent α = 000*1111 = 0^3*1^4 as $(A_0 = \{1, 2, 3\}, A_1 = \{5, 6, 7, 8\})$.

Definition 1.10. The $\textsc{Prefix}_\alpha$ function checks for inputs x that start with minterm α:

$$\textsc{Prefix}_\alpha(x) = \prod_{i \in A_1} x_i \prod_{i \in A_0} (1 - x_i)$$

By now we are used to seeing this notation in the Kronecker delta. The function evaluates to 1 if and only if all of the $x_i \in A_1$ equal 1 and all of the $x_i \in A_0$ equal 0.

Definition 1.11. A shift $\mathrm{s}_j(\alpha)$ is a permutation that shifts α to the right by j bits and adds *s to the beginning. The ith bit of $\mathrm{s}_j(\alpha)$ is given by

$$(\mathrm{s}_j(\alpha))[i] = \begin{cases} \alpha[i-j] & \text{if } i - j > 0 \\ * & \text{otherwise} \end{cases}$$

Definition 1.12. A cyclic shift $\mathrm{cs}_j(\alpha)$ is a permutation that shifts α to the right by j bits modulo the size of the input. That is, the ith bit of $\mathrm{cs}_j(\alpha)$ is given by $(\mathrm{cs}_j(\alpha))[i] = \alpha[(i - j) \bmod n]$.

Example. Given minterm α =10*1 and $n = 8$, we have for example $\mathrm{cs}_1(\alpha) =$ *10*1, $\mathrm{cs}_2(\alpha) =$**10*1, $\mathrm{cs}_5(\alpha) =$1****10*, and $\mathrm{cs}_6(\alpha) =$*1****10.

Definition 1.13. A minterm-cyclic function $\textsc{Match}_\alpha(x)$ is a function that, for a given minterm α and input string x, returns 1 iff x matches some cyclic shift of α. That is:

$$\textsc{Match}_\alpha(x) = \begin{cases} 1 & \text{if } \textsc{Prefix}_{\mathrm{cs}_j(\alpha)}(x) = 1 \text{ for some } j \text{ s.t. } 0 \le j < n \\ 0 & \text{otherwise} \end{cases}$$

Definition 1.14. The *Hamming weight* $|x|$ of $x \in \{0, 1\}^n$ is the number of ones in x. [5]

Example. Given $x = 1010001$ and $y = 0000$, we have $|x| = 3$ and $|y| = 0$.

Definition 1.15. The *size* of a minterm α is the number of 0s, 1s, and *s in the 'reduced' representation of α, wherein we remove the leading and trailing *s.

Example. Let α=*1*10*1**. The 'reduced' representation of α is 1*10*1, so size(α)=6.

Definition 1.16. A *monotonic* (or *monotonically increasing*) function $f : X \to Y$ is one that preserves the order of sets X and Y. For Boolean functions, this ordering is given by Hamming weight, so f is monotone if and only if $f(x) \ge f(y)$ when $|x| \ge |y|$. Equivalently, the value of the function does not decrease when you flip a bit from 0 to 1.

Theorem 1.17. $\textsc{Match}_\alpha$ *is monotone if and only if* $\alpha \in \{1, *\}^k$.

Proof. We provide the intuition behind the proof. For a formal proof of this fact, see [10].
$\leftarrow$ If $\alpha \in \{1, *\}^k$ and we flip a 0 to 1 in x, $\textsc{Match}_\alpha$ will not switch from 1 to 0.
$\rightarrow$ *(Contrapositive)* If α contains a 0 at α_i, consider the input x that matches α at every bit except $x_i = 1$. Then $\textsc{Match}_\alpha(x) = 0$, but if we flip x_i to obtain x', then $\textsc{Match}_\alpha(x') = 1$. Thus, $\textsc{Match}_\alpha$ is not monotone. $\square$

Example. Let α = 11*1. Then $\textsc{Match}_\alpha$ is monotone.

Definition 1.18. A *self-avoiding* minterm α is one where no nontrivial shift of α has a nontrivial intersection with itself.

$$\forall j \text{ s.t. } 0 < j < \text{size}(\alpha),\ \textsc{Prefix}_\alpha(\text{s}_j(\alpha)) = 0$$

Example. $\alpha_1 = 01$, $\alpha_2 = 0^{k_0}1^{k_1}$, $\alpha_3 = 00{*}101$, and $\alpha_4 = 00{*}{*}1101$ are self-avoiding, because $\forall j$ s.t. $0 < j < \text{size}(\alpha),\ \textsc{Prefix}_\alpha(\text{s}_j(\alpha)) = 0$.

Example. $\alpha_5 = 0{*}1$ is not self-avoiding because $\textsc{Prefix}_{\alpha_5}(\text{s}_1(\alpha_5)) = 1$.

Example. $\alpha_6 = 0^{k_0}{*}1^{k_1}$ is not self-avoiding because $\textsc{Prefix}_{\alpha_6}(\text{s}_1(\alpha_6)) = 1$.

Example. $\alpha_7 = 010$ is not self-avoiding because $\textsc{Prefix}_{\alpha_7}(\text{s}_2(\alpha_7)) = 1$.

Example. $\alpha_8 = 0\{0,1\}^k0$ is not self-avoiding because $\textsc{Prefix}_{\alpha_8}(\text{s}_{k+1}(\alpha_8)) = 1$.

For a more detailed look at the classification of self-avoiding minterms, see Chapter 4.

1.3 Polynomial Representation

In the following discussion, we use the convention that cyclic shifts are calculated *modulo* the size n of input string x, but we define the modulus as returning values in $\{1, 2, \cdots, n\}$. Let $\alpha = \{A_0, A_1\}$. Using Definition 1.13, we evaluate the polynomial representing $\textsc{Match}_\alpha$ as the OR of n cyclic shifts $\textsc{Prefix}_{\text{cs}_j(\alpha)}$:

$$\textsc{Match}_\alpha(x) = 1 - \prod_{j=0}^{n-1}(1 - \textsc{Prefix}_{\text{cs}_j(\alpha)}(x)) \tag{1.2}$$

$$\textsc{Match}_\alpha(x) = 1 - \prod_{j=0}^{n-1}(1 - \prod_{i \in A_1} x_{(i+j) \bmod n} \prod_{i \in A_0}(1 - x_{(i+j) \bmod n})) \tag{1.3}$$

That way, if any cyclic shift $\text{cs}_j(\alpha)$ matches the prefix of x, the product will evaluate to 0, so $\textsc{Match}_\alpha(x)$ will evaluate to 1. If no cyclic shift $\text{cs}_j(\alpha)$ matches the prefix of x, the product will evaluate to 1, so $\textsc{Match}_\alpha(x)$ will evaluate to 0.

1.4 Previous Research

Edward Talmage '12 studied the polynomial coefficients of monotone minterm-cyclic functions and proved the following main result:

Theorem 1.19 (Theorem 3.1 of [10])**.** *Let* $\alpha = 1^k = 11\cdots1$ *be a string of* k 1*'s. Let* $f = \text{MATCH}_\alpha(x)$ *be the minterm-cyclic function with minterm* α*. Then deg*$(f) = n$*. Specifically, the coefficient of a degree-*n *term in* p_f *is:* $\begin{cases} -k & \textit{if } (k+1) \mid n \\ 1 & \textit{otherwise} \end{cases}$

Using our terminology, $\text{MATCH}_\alpha(x)$ is *saturated* for α described above.
Talmage also computed degree-n coefficients of all monotone minterm-cyclic functions up to minterms of size 11 and input size $n = 24$. The patterns from these computational results are quite fascinating, and deserve further study and proof. In particular, the following two questions motivate our study:

- For what minterms is $\text{MATCH}_\alpha(x)$ saturated?
- For what minterms are the degree-n coefficients periodic with respect to n?

Chapter 2

Preliminary Results

2.1 Query Complexity and Evasiveness

Theorem 2.1. *All monotone minterm-cyclic functions are evasive.*

Proof. This is a relatively simple application of the topological approach described in [6]. □

This topological approach requires a monotone function, so we cannot apply this result to arbitrary non-monotone minterm-cyclic functions. However, we prove an analogous result for the restricted case of a fixed (or *naïve*, or *non-adaptive*) query algorithm, where the query ordering $\{q_i\}$ is the same for all inputs x. This restriction is equivalent to requiring that each row of the binary decision tree queries a specific bit.

Theorem 2.2. *Given any fixed query ordering $\{q_i\}$ on n elements and for any minterm-cyclic function $\textsc{Match}_\alpha$ (regardless of the monotonicity of α), we can craft an assignment algorithm that requires all n queries to be made.*

Proof. Any nontrivial minterm must contain some 0 or 1. Cyclically shift the minterm α by some j bits such that the last queried element is not a '*'. (That is, $(\mathrm{cs}_j(\alpha))[q_n] = 0$ or 1, where q_n is the index of the last query.)
Then the assignment should answer queries as follows:

$$\begin{cases} 0 & \text{if } \mathrm{cs}_j(\alpha)[q_i] = 0 \\ 1 & \text{if } \mathrm{cs}_j(\alpha)[q_i] = 1 \\ 1 - (\mathrm{cs}_j(\alpha))[q_n] & \text{otherwise} \end{cases}$$

Thus, the number of times $(\mathrm{cs}_j(\alpha))[q_n]$ is assigned previous to the final query is less than the number of occurrences of $(\mathrm{cs}_j(\alpha))[q_n]$ in the minterm, so the querier could not have found an entire minterm.
Also, the answers have not eliminated the possibility of a minterm, because if we answer $(\mathrm{cs}_j(\alpha))[q_n]$ for the last query q_n, then $\textsc{Prefix}_{\mathrm{cs}_j(\alpha)}(x) = 1$ for the chosen j.
Thus, the last query q_n is necessary. □

2.2 Infinite Family of Non-Saturated Monotone Functions

Throughout the following discussion, we maintain our definition that *modulo* returns values in $\{1, 2, \cdots, n\}$.
Previously, Talmage proved via counter-example that not all monotone minterm-cyclic functions are saturated.[10] Here we prove the stronger result that there exist an infinite number of non-saturated monotone minterm-cyclic functions.

Definition 2.3. We define $\chi_S : \{0,1\}^n \to \{0,1\}$ by the product $\chi_S = \prod_{i \in S} x_i$.

Example. $\chi_{\{1,2,4\}} = x_1 x_2 x_4$.

Definition 2.4. We define the *coset* $j + M$ by $\{(j + m) \bmod n : m \in M\}$.

Example. If $n = 4$, then $2+\{1, 2, 4\} = \{(3 \bmod 4), (4 \bmod 4), (6 \bmod 4)\} = \{2, 3, 4\}$.

Definition 2.5. We define *set addition* $T + M$ by $\{(t + m) \bmod n : t \in T, m \in M\}$.

Example. If $n = 4$, then $\{3, 4\} + \{1, 4\} = \{(4 \bmod 4), (7 \bmod 4), (5 \bmod 4), (8 \bmod 4)\} = \{1, 3, 4\}$.

By simplifying Equation (1.3) in the monotone case where $\alpha = (\varnothing, A_1)$,

$$\textsc{Match}_\alpha(x) = 1 - \prod_{j=0}^{n-1}(1 - \prod_{i \in A_1} x_{(i+j) \bmod n})$$

Or, more compactly:

$$\textsc{Match}_\alpha(x) = 1 - \prod_{j=0}^{n-1}(1 - \chi_{j+A_1}) \tag{2.1}$$

If we were to expand out this product in Equation (2.1), we would get 2^n terms, each of which corresponds to choosing some subset $T \subseteq [n]$ of values j from the product. This term will include every term x_i from x if and only if $T + A_1 \supseteq [n]$, or equivalently, $T + A_1 = [n]$ (since set addition is defined modulo n). The coefficient of this term will depend upon the parity of the size of T: $(-1)^{|T|+1}$. Thus, the degree-n coefficient of $\textsc{Match}_\alpha$ will be:

$$\sum_{\substack{T \subseteq [n], \\ T + A_1 = [n]}} (-1)^{|T|+1} = -\sum_{\substack{T \subseteq [n], \\ T + A_1 = [n]}} (-1)^{|T|} \tag{2.2}$$

Theorem 2.6. *Given non-saturated monotone minterm-cyclic function* $\textsc{Match}_\alpha$ *(such that* $\deg(\textsc{Match}_\alpha) < n$ *for input size* n*), there exists an infinite family of minterm-cyclic functions that are not saturated.*

Proof. Equation (2.2) gives us the equation for the degree-n coefficient of the polynomial representing $\textsc{Match}_\alpha$:

$$-\sum_{\substack{T \subseteq [n], \\ T + A_1 = [n]}} (-1)^{|T|} = 0$$

For any positive integer k, create minterm $\alpha' = (\varnothing, A'_1)$, where $A'_1 = \{km : m \in A_1\}$. We claim that on input size kn, deg($\textsc{Match}_{\alpha'}) < kn$.

Consider any set $T' \subseteq [kn]$ such that $T' + A'_1 = [kn]$.
For every j s.t. $0 \leq j < k$, let $T_j := \{t \in T' : t \bmod k = j\}$
It should be clear that all T_j are disjoint, and that $T' = \bigcup_j T_j$.
Thus,

$$T' + A'_1 = \{(t+m) \bmod n : t \in \bigcup_j T_j, m' \in A'_1\}$$
$$T' + A'_1 = \bigcup_j \{(t+m) \bmod n : t \in T_j, m' \in A'_1\}$$
$$[kn] = T' + A'_1 = \bigcup_j (T_j + A'_1)$$

Subclaim 1. $0 \leq i < j < k \Rightarrow (T_i + A'_1) \cap (T_j + A'_1) = \varnothing$
For any $j \in [k]$, each $t' \in T_j$ must satisfy $t' = kt + j$ for some integer $t \in [n]$.
If $x \in (T_i + M') \cap (T_j + A'_1)$, then $x = (kt_1 + i) + kA_1 = (kt_2 + j) + kA_2$
Thus, $x \bmod k = i = j$, but $i < j < k$, so this is a contradiction.

Let $\hat{T}_j = \{t \in [n] : kt + j \in T_j\}$
Then $\hat{T}_j + A_1 = \{t + m : kt + j \in T_j, km \in A'_1\}$

Subclaim 2. $T_j + A'_1 \supseteq \{ki + j : 1 \leq i \leq n\} \Leftrightarrow \hat{T}_j + A_1 \supseteq [n]$
The following form a logical chain of equivalences, based on the definitions above:

$$\begin{aligned} & T_j + A'_1 \supseteq \{ki + j : 0 \leq i < n\} \\ \Leftrightarrow\ & \forall\, 0 \leq i < n\ \ \exists(kt + j) \in T_j, \exists(km) \in A'_1 \text{ s.t. } (t+m) \bmod k \equiv i \\ \Leftrightarrow\ & \forall\, 0 \leq i < n\ \ \exists(t) \in \hat{T}_j, \exists(m) \in A_1 \text{ s.t. } (t+m) \bmod k \equiv i \\ \Leftrightarrow\ & \hat{T}_j + A_1 \supseteq \{i : 0 \leq i < n\} \end{aligned}$$

Subclaim 3. $T' + A'_1 \supseteq [kn] \Leftrightarrow T' = \bigcup_j T_j$, where each $\hat{T}_j$ satisfies $\hat{T}_j + A_1 \supseteq [n]$.
Using *Subclaim 1* and *Subclaim 2*, the following form a chain of logical equivalences:

$$\begin{aligned} & T' + A'_1 \supseteq [kn] \\ \Leftrightarrow\ & \forall\, 0 \leq j < k\ \ T_j + A'_1 \supseteq \{ki + j : 0 \leq i < n\} \\ \Leftrightarrow\ & \forall\, 0 \leq j < k\ \ \hat{T}_j + A_1 \supseteq [n] \\ \Leftrightarrow\ & T' = \bigcup_j T_j, \text{ where each } \hat{T}_j \text{ satisfies } \hat{T}_j + A_1 \supseteq [n] \end{aligned}$$

Subclaim 3 tells us that any combination of $\hat{T}_j$ that cover $[n]$ will give a set T' s.t. $T' + A'_1 = [kn]$, and that these are the only such sets T'. Furthermore, since these sets T_j

are disjoint, any distinct combination of these sets will yield a distinct set T'. Finally, recall our assumption that deg(MATCH$_\alpha$) $< n$, which implies by Equation (2.2) that:

$$-\sum_{\substack{\hat{T}_j \subseteq [n], \\ \hat{T}_j + A_1 = [n]}} (-1)^{|\hat{T}_j|} = 0$$

for all j. Therefore,

$$\sum_{\substack{T' \subseteq [kn], \\ T' + A_1 = [kn]}} (-1)^{|T'|} = \sum_{j=0}^{n-1} (\sum_{\substack{\hat{T}_j \subseteq [n], \\ \hat{T}_j + A_1 = [n]}} (-1)^{|T_j|}) = \sum_{j=0}^{n-1} (0) = 0$$

Thus, the degree-kn coefficient of the polynomial representing MATCH$_{\alpha'}$ is 0, so MATCH$_{\alpha'}$ is not saturated. We can build an infinite family of such functions MATCH$_{\alpha'}$ by repeating the above 'expansion' process for every positive integer k. This proves our claim. □

Chapter 3

Evaluation of Polynomial Degree

3.1 Using Parity to Evaluate Polynomial Degree

Here we expand the multilinear polynomial representing $f : \{0,1\}^n \rightarrow \{0,1\}$ in order to evaluate its polynomial degree using the parity of satisfying x. Recall that we can represent f by interpolating its value at each of the 2^n points in its domain $\{0,1\}^n$. We reproduce Equation (1.1) here:

$$p_f(x) = \sum_{\alpha \in \{0,1\}^n} \delta_\alpha(x) f(\alpha) = \sum_{\substack{\alpha \in \{0,1\}^n \\ f(\alpha)=1}} \delta_\alpha(x)$$

At this point, we can evaluate $\delta_\alpha(x)$ as the AND of XNORs.[2]

$$\delta_\alpha(x) = \text{AND}\Big(\text{XNOR}(\alpha_1, x_1), \text{XNOR}(\alpha_2, x_2), ..., \text{XNOR}(\alpha_n, x_n)\Big)$$
$$\delta_\alpha(x) = \prod_{i=1}^{n} (\alpha_i x_i + (1-\alpha_i)(1-x_i))$$
$$\delta_\alpha(x) = \prod_{i=1}^{n} ((2\alpha_i - 1)x_i + (1-\alpha_i))$$
$$\delta_\alpha(x) = \prod_{i=1}^{n} ((-1)^{\alpha_i+1} x_i + (1-\alpha_i))$$

From this last equation, the coefficient of $\chi_{[n]}$ in the expansion of δ_α is:

$$\prod_{i=1}^{n} ((-1)^{\alpha_i+1}) = (-1)^n (-1)^{|\alpha|}$$

Interchangeably, we can also replace the Hamming weight $|\alpha|$ with the *parity* of α, which is 1 iff $|\alpha|$ is odd.

[2]On two bits of input, XNOR tests for equality. That is, XNOR(0,1) = XNOR(1,0) = 0, while XNOR(0,0) = XNOR(1,1) = 1.

We substitute this into Equation (1.1) to get the coefficient of $\chi_{[n]}$ in f:

$$f_{[n]} = \sum_{\substack{\alpha\in\{0,1\}^n \\ f(\alpha)=1}} (-1)^n(-1)^{|\alpha|} = (-1)^n \sum_{\substack{\alpha\in\{0,1\}^n \\ f(\alpha)=1}} (-1)^{|\alpha|} = (-1)^n\Big(\sum_{\substack{\alpha\in\{0,1\}^n \\ f(\alpha)=1 \\ |\alpha| \text{ even}}} 1 - \sum_{\substack{\alpha\in\{0,1\}^n \\ f(\alpha)=1 \\ |\alpha| \text{ odd}}} 1\Big)$$

We define sets $S_{\text{sat}}^{\text{even}}, S_{\text{sat}}^{\text{odd}}, S_{\text{unsat}}^{\text{even}}, S_{\text{unsat}}^{\text{odd}}$ to simplify this notation, where 'sat' represents x that *satisfies* f.

$$\begin{aligned} S_{\text{sat}}^{\text{even}} &= \{\alpha : |\alpha| \text{ even}, f(\alpha) = 1\} \\ S_{\text{sat}}^{\text{odd}} &= \{\alpha : |\alpha| \text{ odd}, f(\alpha) = 1\} \\ S_{\text{unsat}}^{\text{even}} &= \{\alpha : |\alpha| \text{ even}, f(\alpha) = 0\} \\ S_{\text{unsat}}^{\text{odd}} &= \{\alpha : |\alpha| \text{ odd}, f(\alpha) = 0\} \end{aligned}$$

Then the above equation for $f_{[n]}$ simplifies to:

$$f_{[n]} = (-1)^n(|S_{\text{sat}}^{\text{even}}| - |S_{\text{sat}}^{\text{odd}}|)$$

Quick Lemma: $\forall n > 0 \ \sum_{k \text{ odd}} \binom{n}{k} = \sum_{k \text{ even}} \binom{n}{k}$
By the Binomial Theorem,

$$\sum_{\substack{k \text{ odd} \\ 0\le k\le n}} \binom{n}{k} - \sum_{\substack{k \text{ even} \\ 0\le k\le n}} \binom{n}{k} = \sum_{k=0}^{n} (-1)^k \binom{n}{k} = (1 + (-1))^n = 0^n = 0 \text{ for } n > 0.$$

By this lemma, we conclude:

$$|S_{\text{sat}}^{\text{even}}| + |S_{\text{unsat}}^{\text{even}}| = |S_{\text{sat}}^{\text{odd}}| + |S_{\text{unsat}}^{\text{odd}}|$$

$$\therefore f_{[n]} = (-1)^n(|S_{\text{sat}}^{\text{even}}| - |S_{\text{sat}}^{\text{odd}}|) = (-1)^n(|S_{\text{unsat}}^{\text{odd}}| - |S_{\text{unsat}}^{\text{even}}|) \tag{3.1}$$

Evaluating the equation is the main method we will use in the following sections to prove results about degree-n coefficients and saturated functions. Note that a quicker but less comprehensible version of the above result relies on the Möbius Inversion Formula.[1, 2]

3.2 Fourier Expansion

It's worth noting that one can also use the Fourier expansion of a Boolean function to evaluate the coefficients of the multilinear polynomial representing a Boolean function. In this type of analysis, instead of representing a Boolean function as $f : \{0,1\}^n \to \{0,1\}$, one can use $f' : \{-1,1\}^n \to \{-1,1\}$. This notation has a more obvious connection to the parity function of Section 3.1, because of the property that $(x')^2 = 1 \ \forall x' \in \{-1,1\}$.

We define the *correlation* between two functions $f, g : \{-1,1\}^n \to \mathbb{R}$ as the expected value of their product:

$$\begin{aligned} \mathrm{Cor}(f,g) &= \qquad \mathbb{E}[f(x)g(x)] \\ \mathrm{Cor}(f,g) &= \sum_{x\in\{-1,1\}^n} f(x)g(x)\cdot 2^{-n} \end{aligned}$$

This forms a positive definite inner product on the space of Boolean functions. Using this formulation, we can obtain similar results to those in Section 3.1. See Ryan O'Donnell's tutorial on Boolean functions for a more complete introduction to this type of analysis. [7]

3.3 Operations on the Bits of a Minterm

Here we define three basic and useful operations on the bits of a minterm, which will be useful in the proofs to follow.

Definition 3.1. Given a minterm $\alpha \in \{0,1,*\}^n$, denote its bitwise (1's) complement by $\overline{\alpha} \in \{0,1,*\}^n$, where each 0 is flipped to 1 and each 1 is flipped to 0.

Definition 3.2. Given a minterm $\alpha \in \{0,1,*\}^n$, we use $\hat{\alpha}$ to denote the minterm where the first bit of α is replaced by a '*' and every subsequent bit remains the same.

$$\hat{\alpha}[i] = \begin{cases} * & \text{if } i = 1 \\ \alpha[i] & \text{otherwise} \end{cases}$$

Some readers will note that this notation $\hat{\alpha}$ clashes with the notation for Fourier coefficients $\hat{f}(S)$. However, it should always be clear from context whether this notation $\hat{\cdot}$ is applied to a minterm or a function.

Definition 3.3. Given a minterm $\alpha \in \{0,1,*\}^n$, we use $\tilde{\alpha}$ to denote the minterm where the first bit of α is flipped (replaced by $1-\alpha[1]$) and every subsequent bit remains the same.

$$\tilde{\alpha}[i] = \begin{cases} 1-\alpha[i] & \text{if } i = 1 \\ \alpha[i] & \text{otherwise} \end{cases}$$

Theorem 3.4. *For $\alpha \in \{0,1,*\}^k$, define $f := \textsc{Match}_\alpha$ and $f' := \textsc{Match}_{\overline{\alpha}}$. Then:*

$$f_{[n]} = (-1)^n f'_{[n]}. \tag{3.2}$$

Proof. Note that $\textsc{Match}_\alpha(x) = 1 \Leftrightarrow \textsc{Match}_{\overline{\alpha}}(\overline{x}) = 1$. To evaluate the degree-n coefficient of $\textsc{Match}_{\overline{\alpha}}$, we use Equation (3.1). If n is even, the parity of x is the same as the parity of $\overline{x}$, so the degree-n coefficients of $\textsc{Match}_{\overline{\alpha}}$ and $\textsc{Match}_\alpha$ are equal. If n is odd, the parity of $\overline{x}$ is opposite that of x, so the degree-n coefficient of $\textsc{Match}_{\overline{\alpha}}$ is -1 times the degree-n coefficient of $\textsc{Match}_\alpha$. □

3.4 Examples

3.4.1 The Minterm $\alpha = 01$

For convenience of notation, let $f = \textsc{Match}_\alpha$.

Theorem 3.5. *Nonzero degree-n coefficients $f_{[n]}$ are periodic with period 2:*

$$f_{[n]} = \begin{cases} -2 & \textit{if } n \textit{ even} \\ 0 & \textit{if } n \textit{ odd} \end{cases}$$

Proof. The only nonsatisfying assignments are: $x_0 = 0^n$, $x_1 = 1^n$.
When input size n is even, both x_0 and x_1 have $|x|$ even.
$\Rightarrow |S^{\text{even}}_{\text{unsat}}| = 2$, $|S^{\text{odd}}_{\text{unsat}}| = 0$, so by Equation (3.1) above, $f_{[n]} = -2$.
When input size n is odd, $|x_0|$ is even and $|x_1|$ is odd.
$\Rightarrow |S^{\text{even}}_{\text{unsat}}| = |S^{\text{odd}}_{\text{unsat}}| = 1$, so by Equation (3.1) above, $f_{[n]} = 0$. □

3.4.2 Self-Avoiding Minterms

Note that this is a generalization of Subsection 3.4.1, since $\alpha = 01$ is self-avoiding.
For convenience of notation, let $f = \textsc{Match}_\alpha$.

Theorem 3.6. *For $\alpha \in \{0, 1, *\}^k$ self-avoiding containing at least one *: $f_{[n]} = 0 \;\; \forall\, n$. For $\alpha \in \{0, 1\}^k$ self-avoiding:*

$$f_{[n]} = \begin{cases} \pm \textit{size}(\alpha) & \textit{if size}(\alpha) \mid n \\ 0 & \textit{otherwise} \end{cases}$$

Proof. Recall that $\hat{\alpha}$ is α with the first bit replaced by '*'. We define a function CONVERT:

Algorithm 1 CONVERT(x)

Input: $x \in \{0, 1\}^n$ s.t. $f(x) = 1$
$i_0 :=$ first bit of the 1st occurrence of α
$i := \big(i_0 + \text{size}(\alpha)\big) \bmod \text{size}(\alpha)$
$j := 0$
while $j < \frac{n}{\text{size}(\alpha)} - 1$ **do**
    if $(\textsc{Prefix}_{\hat{\alpha}}(\text{cs}_{-i}(x)) == 0)$ **or** $(i > i_0)$ **then**
        return x with bit i flipped
    else
        $i := \big(i + \text{size}(\alpha)\big) \bmod \text{size}(\alpha)$
        $j := j + 1$
    end if
end while
return FAILURE

The intuition behind this proof is that, in most cases, CONVERT is a bijection between $S^{\text{even}}_{\text{sat}}$ and $S^{\text{odd}}_{\text{sat}}$. Specifically, CONVERT attempts to flip a single bit of the input without altering the first occurrence of the minterm α or creating an earlier occurrence of α.

Subclaim 1. For $x \in \{0,1\}^n$ s.t. $\textsc{Convert}(x) \neq \textsc{Failure}$: if $f(x) = 1$, then $f(\textsc{Convert}(x)) = 1$.

$\textsc{Convert}$ finds the first occurrence of the minterm, and the bit it flips is $(j \cdot \text{size}(\alpha))+1$ bits after the end of this minterm, for some j s.t. $0 \leq j < \frac{n}{\text{size}(\alpha)} - 1$. Combining these, the bit flipped is k bits after the end of the first occurrence of α, where $1 \leq k < n - \text{size}(\alpha)+1$. The first bit of that first occurrence of α is by definition $n - \text{size}(\alpha) + 1$ bits after the end of that occurrence of α, so the bit we flip cannot be within that occurrence of α. Therefore, $f(\textsc{Convert}(x)) = 1$.

Subclaim 2. $\textsc{Convert}$ does not create any new occurrence of minterm α that starts before i_0.

Assume to the contrary that $\textsc{Convert}$ creates a new occurrence of α that starts before i_0. Then the flipped bit i must be in the translated minterm. This minterm must start k bits before i, for some $0 \leq k \leq \text{size}(\alpha) - 1$. If $k > 0$, then:

$$\textsc{Prefix}_{\hat{\alpha}}(s_{\text{size}(\alpha)-k}(\alpha)) = 1$$

$$\textsc{Prefix}_{\alpha}(s_{\text{size}(\alpha)-k}(\alpha)) = 1$$

So α is not self-avoiding.
This leaves the possibility that $k = 0$: that is, that the new occurrence of α starts at bit i. Since $i > i_0$, in order to **return**, the **if** statement must have evaluated $\textsc{Prefix}_{\hat{\alpha}}(\text{cs}_{-i}(x)) == 0$, so the bits directly following i do not match $\hat{\alpha}$. Thus, the new occurrence of the minterm could not have started at bit i. This is a contradiction, so it must be that $\textsc{Convert}$ does not create any new occurrence of minterm α that starts before i_0.

Subclaim 3. Let $X = \{x \in \{0,1\}^n \text{ s.t. } \textsc{Convert}(x) \neq \textsc{Failure}\}$.
Then $\textsc{Convert}$ is an involution from X to X.

We want to show that $\textsc{Convert}(\textsc{Convert}(x)) = x \;\; \forall x \in X$.
Assume to the contrary that $\textsc{Convert}(\textsc{Convert}(x)) \neq x$. By *Subclaim 2,* we do not create any new occurrence of the minterm that starts before i_0, so i_0 is the start of the first occurrence of α in both x and $\textsc{Convert}(x)$. Then let i be the first differing bit after i_0. It must have been flipped in $\textsc{Convert}$ on either input x or input $\textsc{Convert}(x)$, but not both.
Say it was flipped in $\textsc{Convert}$ on input x. Then when we run $\textsc{Convert}$ on input $\textsc{Convert}(x)$, each **if** statement should evaluate the same way, and we should flip the same bit. Thus, this is a contradiction.
Say it was flipped in $\textsc{Convert}$ on input $\textsc{Convert}(x)$. Then $\textsc{Prefix}_{\hat{\alpha}}(\text{cs}_{-i}(x)) = 0$ but $\textsc{Prefix}_{\hat{\alpha}}(\text{cs}_{-i}(\textsc{Convert}(x))) = 1$. The only bit that changed between x and $\textsc{Convert}(x)$ is bit i, which does not affect either instance of $\textsc{Prefix}$ above. This is a contradiction.
Therefore, $\textsc{Convert}$ is an involution from X to X.

Also note that $\textsc{Convert}(x)$ must have one bit flipped from x: the parity of $\textsc{Convert}(x)$ is different from the parity of x. That is, $x \in S_{\text{sat}}^{\text{even}} \Rightarrow \textsc{Convert}(x) \in S_{\text{sat}}^{\text{odd}}$ and $x \in$

$S_{\text{sat}}^{\text{odd}} \Rightarrow \textsc{Convert}(x) \in S_{\text{sat}}^{\text{even}}$. Then CONVERT is a bijection from $X \cap S_{\text{sat}}^{\text{even}}$ to $X \cap S_{\text{sat}}^{\text{odd}}$.

$$\therefore |X \cap S_{\text{sat}}^{\text{even}}| = |X \cap S_{\text{sat}}^{\text{odd}}| \tag{3.3}$$

Case 1: size$(\alpha) \nmid n$

Subclaim 4. CONVERT is well-defined on $x \in \{0,1\}^n$ s.t. $f(x) = 1$, and will never return FAILURE on any such x.

First, note that the **while** loop can only run a finite number of times before CONVERT returns. Then it suffices to show that CONVERT will never return FAILURE when size$(\alpha) \nmid n$. Assume to the contrary that CONVERT returns FAILURE on some input x. Then the algorithm must have gone through $\lceil \frac{n}{\text{size}(\alpha)} - 1 \rceil$ iterations of **while**. During each of these iterations, it increments i by size$(\alpha) \bmod n$, so before the final iteration, $0 < (i_0 - i) \bmod n <$ size(α). However, we also know that x matches α beginning at i_0 and $\hat{\alpha}$ beginning at i:

$$\textsc{Prefix}_{\hat{\alpha}}(\text{cs}_{-i}(x)) == 1 \qquad \text{and} \qquad \textsc{Prefix}_{\alpha}(\text{cs}_{-i_0}(x)) == 1.$$

Combining these, we notice that this implies:

$$\textsc{Prefix}_{\alpha}(\text{s}_{i_0 - i}(\alpha)) == 1.$$

This contradicts the assumption that α is self-avoiding. Therefore, it cannot be that CONVERT returns FAILURE on any input x.

By Equation (3.3), this implies that $|S_{\text{sat}}^{\text{even}}| = |S_{\text{sat}}^{\text{odd}}|$.
By the above claims and Equation (3.1), we conclude that $f_{[n]} = 0$.

Case 2: size$(\alpha) \mid n$
Recall that $\tilde{\alpha}$ is defined as α with the first bit flipped.

Then let $S = \{\text{cs}_i(\tilde{\alpha}^{r-1}\alpha) : 0 \leq i < \text{size}(\alpha)\}$, where $r = \frac{n}{\text{size}(\alpha)}$.

Subclaim 5. CONVERT is well-defined on $x \in \{0,1\}^n$ s.t. $f(x) = 1$ and $x \notin S$, and CONVERT will not return FAILURE on any such x.

Again, note that the **while** loop can only run a finite number of times before CONVERT returns. It suffices to show that CONVERT will not return FAILURE on x described above.

Assume to the contrary that CONVERT returns FAILURE on some input x. For each of the $\frac{n}{\text{size}(\alpha)} - 1$ iterations of **while**, we have that $i < i_0$ and $(\textsc{Prefix}_{\hat{\alpha}}(\text{cs}_{-i}(x)) == 1)$. Further, if $(\textsc{Prefix}_{\alpha}(\text{cs}_{-i}(x)) == 1)$ in any iteration of the **while** loop with $i < i_0$, this contradicts the assumption that i_0 is the first bit of the first occurrence of α. Therefore, the only x that satisfy these stipulations must equal some cyclic shift cs$_i$ of $\overline{\alpha}^{r-1}\alpha$, for $0 \leq i <$ size(α). All such x are in S, and are excluded from the subclaim, so the claim is true.

By Equation (3.3), this implies that $|S_{\text{sat}}^{\text{even}} - S| = |S_{\text{sat}}^{\text{odd}} - S|$.
This allows us to evaluate Equation (3.1) as follows:

$$f_{[n]} = (-1)^{n-1}(|S \cap S_{\text{sat}}^{\text{odd}}| - |S \cap S_{\text{sat}}^{\text{even}}|) \tag{3.4}$$

Now we evaluate the parity and number of $x \in S$:
First, note that if α contains a '*', then the number of $x \in S$ of even parity is the same as the number of $x \in S$ of odd parity. In this case, $f_{[n]} = (-1)^{n-1}(|S \cap S_{\text{sat}}^{\text{odd}}| - |S \cap S_{\text{sat}}^{\text{even}}|) = 0$. This proves the first half of Theorem 3.6.

If $\alpha \in \{0,1\}^k$, then every $x \in S$ has the same parity, and $|S| = \text{size}(\alpha)$. In this case, by Equation (3.4),

$$f_{[n]} = (-1)^{n-1}((-1)^{|x|+1}|S|) = (-1)^{n+|x|}\text{size}(\alpha).$$

More specifically, the parity of $x \in S$ is:

$$|x| = |\alpha| + |\hat{\alpha}|(r-1) = |\alpha| + (|\alpha|+1)(r-1) = r(|\alpha|+1) - 1.$$

$$f_{[n]} = (-1)^{n+|x|}\text{size}(\alpha) = (-1)^{n+r(|\alpha|+1)-1}\text{size}(\alpha).$$

We conclude that in this case $f_{[n]} = (-1)^{n+r(|\alpha|+1)-1}\text{size}(\alpha)$, proving Theorem 3.6. □

3.4.3 The Minterm $\alpha = 1^k$

This is the same minterm studied previously by Talmage [10]. Given our formulation, the proof is quicker. For convenience of notation, let $f = \text{MATCH}_\alpha$.

Theorem 3.7. *f is saturated, with the following degree-n coefficients:*

$$f_{[n]} = \begin{cases} -k & \textit{if } (k+1) \mid n \\ 1 & \textit{otherwise} \end{cases}$$

Proof. Given $\alpha = 1^k$, let $\alpha' = 1^k0$. Let $f = \text{MATCH}_\alpha$, $g = \text{MATCH}_{\alpha'}$, and $h = \text{MATCH}_0$. By substituting $f(x)$ with the equivalent $f(x) = f(x)g(x) + f(x)(1-g(x))$, we deduce:

$$\text{MATCH}_\alpha(x) = \big(\text{MATCH}_\alpha(x)\text{MATCH}_{\alpha'}(x)\big) + \big(\text{MATCH}_\alpha(x)(1 - \text{MATCH}_{\alpha'}(x))\big)$$

We notice that the first term $\text{MATCH}_\alpha(x)\text{MATCH}_{\alpha'}(x)$ simplifies to $\text{MATCH}_{\alpha'}(x)$, since $\text{MATCH}_{\alpha'}(x) \Rightarrow \text{MATCH}_\alpha(x)$.
We can also simplify the second product. Notice that (x contains 1^k and contains a 0 anywhere) $\Leftrightarrow$ (x contains 1^k0). Then $\text{MATCH}_\alpha(x)(1 - \text{MATCH}_{\alpha'}(x)) = 1 - \text{MATCH}_0(x)$.
Now our identity simplifies to:

$$\text{MATCH}_\alpha(x) = \text{MATCH}_{\alpha'}(x) + 1 - \text{MATCH}_0(x)$$

Using our notation from above, $f = g - h + 1$, so all the coefficients $f_{[n]} = g_{[n]} - h_{[n]}$. Notice that α' is self-avoiding, so by Theorem 3.6,

$$g_{[n]} = \begin{cases} (-1)^{n+\frac{n}{k+1}(k+1)-1}(k+1) & \text{if } (k+1) \mid n \\ 0 & \text{otherwise} \end{cases}$$

Simplifying the leading term, we get:

$$g_{[n]} = \begin{cases} -(k+1) & \text{if } (k+1) \mid n \\ 0 & \text{otherwise} \end{cases}$$

And by examination, we see that $h_{[n]} = -1 \; \forall \; n$.

We conclude with the result:

$$f_{[n]} = \begin{cases} -k & \text{if } (k+1) \mid n \\ 1 & \text{otherwise} \end{cases}$$

□

3.4.4 The Minterm $\alpha = 0 * 1$

For convenience of notation, let $f = \textsc{Match}_\alpha$.

Theorem 3.8. *Nonzero degree-n coefficients are periodic with period 4:*

$$f_{[n]} = \begin{cases} -4 & \textit{if } 4 \mid n \\ 0 & \textit{otherwise} \end{cases}$$

Proof. The only nonsatisfying assignments are:

- $x_0 = 0^n$
- $x_1 = 1^n$
- $x_{\text{alt}_0} = (01)^{n/2}$ if $|n|$ even
- $x_{\text{alt}_1} = (10)^{n/2}$ if $|n|$ even

When input size n is odd, $|x_0|$ is even and $|x_1|$ is odd.

$\Rightarrow |S^{\text{even}}_{\text{unsat}}| = |S^{\text{odd}}_{\text{unsat}}| = 1$, so by Equation (3.1) above, $f_{[n]} = 0$.

When $2 \mid n$ but $4 \nmid n$, $|x_0|$ and $|x_1|$ are even, and $|x_{\text{alt}_0}|$ and $|x_{\text{alt}_1}|$ are odd.

$\Rightarrow |S^{\text{even}}_{\text{unsat}}| = |S^{\text{odd}}_{\text{unsat}}| = 2$, so by Equation (3.1) above, $f_{[n]} = 0$.

When $4 \mid n$, then $|x_0|$, $|x_1|$, $|x_{\text{alt}_0}|$, and $|x_{\text{alt}_1}|$ are even.

$\Rightarrow |S^{\text{even}}_{\text{unsat}}| = 4$, $|S^{\text{odd}}_{\text{unsat}}| = 0$, so by Equation (3.1) above, $f_{[n]} = -4$.

□

3.4.5 Minterms that Self-Overlap Only at the First Shift

Note that this generalizes Subsection 3.4.4, since $\alpha = 0$*1 self-overlaps only at the first shift s_1.

We assume that $\textsc{Prefix}_\alpha(\text{s}_1(\alpha)) = 1$ and $\forall j$ s.t. $1 < j <$ size(α), $\textsc{Prefix}_\alpha(\text{s}_j(\alpha)) = 0$.

For convenience of notation, let $f = \textsc{Match}_\alpha$.

Hypothesis 3.9. If $\alpha \in \{0,1,*\}^k$ s.t. α contains the pattern '**', then $f_{[n]} = 0 \ \forall n$. If $\alpha \in \{0,1,*\}^k$ s.t. α does not contain the pattern '**', then

$$f_{[n]} = \begin{cases} \pm(\text{size}(\alpha)+1) & \text{if } (\text{size}(\alpha)+1) \mid n \\ 0 & \text{otherwise} \end{cases}$$

3.4.6 Minterms that Self-Overlap Only at the ith Shift

Note that this further generalizes Subsection 3.4.5.

Hypothesis 3.10. If a minterm α self-overlaps at only one shift s_i, then create a new minterm α' by evaluating $\text{OR}(\alpha, \mathrm{s}_i(\alpha))$. If α' is self-avoiding, then MATCH_α has the same degree-n coefficients as $\text{MATCH}_{\alpha'}$ (up to negation).

3.4.7 The Minterm $\alpha = 11 * 1$

For convenience of notation, let $f = \text{MATCH}_\alpha$.

Theorem 3.11. MATCH_α *is not saturated; specifically,* $f_{[n]} = 0$ *for* $n = 12$.

Proof. Currently, we have two proofs of this fact. The first is a computer-assisted proof using Equation (2.2), first generated by Talmage [10]. The second is an exhaustive case-based analysis of sets T s.t. $T + \{1,2,4\} = [12]$, generated by hand. This includes cases from $5 \le |T| \le 12$, and concludes by evaluating the sum $1-12+132-208+387-384+108-24 = 0$ to calculate the degree-n coefficient. We are in the process of formalizing a bijective proof using Equation (3.1). □

Chapter 4

Characterization of Self-Avoiding Minterms

As we saw at the end of Chapter 3, the characterization of a minterm as self-avoiding (or the characterization of its self-overlaps) can lead to interesting results about the polynomial complexity of the associated minterm-cyclic function. Given this motivation, this chapter serves to collect and generalize certain patterns in self-avoiding or self-overlapping minterms. In most small cases below, we provide the necessary intuition without formal proof.

4.1 Self-Avoiding Minterms

Theorem 4.1. *The minterm $\alpha = 0^{k_0}1^{k_1}$ is self-avoiding for any $k_0, k_1 > 0$.*

Proof. This is intuitive, but we will prove this more rigorously because of its similarity to the examples to follow.
The minterm α is defined as:

$$\alpha[i] = \begin{cases} 0 & \text{if } 0 < i \leq k_0 \\ 1 & \text{if } k_0 < i \leq k_0 + k_1 \end{cases}$$

Assume to the contrary that for some j s.t. $0 < j < k_0 + k_1$, $\textsc{Prefix}_\alpha(\mathrm{s}_j(\alpha)) = 1$. If $k_0 < j < k_0 + k_1$, then $\alpha[j+1] = 1$ and $\mathrm{s}_j(\alpha)[j+1] = \alpha[j+1-j] = \alpha[1] = 0$. This is a contradiction. If $0 < j \leq k_0$, then $\alpha[k_0+1] = 1$ and $\mathrm{s}_j(\alpha)[k_0+1] = \alpha[k_0+1-j] = 0$. This is also a contradiction. Therefore, we conclude that α is self-avoiding. $\square$

Theorem 4.2. *The minterm $\alpha = 0^{k_0}1^{k_1}0^{k_2}1^{k_3}$ is self-avoiding for $k_0, k_1, k_2, k_3 > 0$ s.t. $k_0 > k_2$ or $k_1 < k_3$.*

Proof. The minterm α is defined as:

$$\alpha[i] = \begin{cases} 0 & \text{if } 0 < i \leq k_0 \\ 1 & \text{if } k_0 < i \leq k_0 + k_1 \\ 0 & \text{if } k_0 + k_1 < i \leq k_0 + k_1 + k_2 \\ 1 & \text{if } k_0 + k_1 + k_2 < i \leq k_0 + k_1 + k_2 + k_3 \end{cases}$$

Assume to the contrary that for some j s.t. $0 < j < k_0+k_1+k_2+k_3$, $\text{PREFIX}_\alpha(\text{s}_j(\alpha)) = 1$.
Case 1: If $0 < j \leq k_0$, then $\alpha[k_0+1] = 1$ and $\text{s}_j(\alpha)[k_0+1] = \alpha[k_0+1-j] = 0 \Rightarrow$ contradiction.
Case 2: If $k_0 < j \leq k_0+k_1$, or $k_0+k_1+k_2 < j < k_0+k_1+k_2+k_3$, then $\alpha[j+1] = 1$ and $\text{s}_j(\alpha)[j+1] = \alpha[1] = 0 \Rightarrow$ contradiction.
Case 3: If $k_0+k_1 < j \leq k_0+k_1+k_2$ and $k_0 > k_2$, then $\alpha[k_0+k_1+k_2+1] = 1$ and $\text{s}_j(\alpha)[k_0+k_1+k_2+1] = \alpha[k_0+k_1+k_2+1-j] = 0$ because $k_0+k_1+k_2+1-j < k_2+1 < k_0+1 \Rightarrow$ contradiction.
Case 4: If $k_0+k_1 < j \leq k_0+k_1+k_2$ and $k_1 < k_3$:
$\alpha[k_0+k_1+k_2] = 0 \Rightarrow \text{s}_j(\alpha)[k_0+k_1+k_2] = \alpha[k_0+k_1+k_2+1-j] = 0$.
$\alpha[k_0+k_1+k_2+1] = 1 \Rightarrow \text{s}_j(\alpha)[k_0+k_1+k_2+1] = \alpha[k_0+k_1+k_2+1-j] = 1$.
Then $1 \leq k_0+k_1+k_2-j \leq k_0$ and $k_0 < k_0+k_1+k_2-j+1 \leq k_0+k_2$. By some algebra, $k_1+k_2 \leq j < k_1+k_2+1$, so $j = k_1+k_2$.
But then $0 = \alpha[k_0+k_1+1] = \text{s}_j(\alpha)[k_0+k_1+1+j] = \alpha[k_0+k_1+1+j] = 0$, but $k_2 < j = k_1+k_2 < k_2+k_3$, so $k_0+k_1+k_2+1 < k_0+k_1+1+j < k_0+k_1+k_2+k_3$, so $\alpha[k_0+k_1+k_2+1-j] = 0 \Rightarrow$ contradiction.
This covers all cases. Therefore, we conclude that α is self-avoiding. □

Theorem 4.3. *The minterm* $\alpha = 0^{k_0^0}1^{k_1^0}0^{k_0^1}1^{k_1^1}0^{k_0^2}1^{k_1^2}$ *is self-avoiding for* $k_0^0, k_1^0, k_0^1, k_1^1, k_0^2, k_1^2 > 0$ *s.t.:*

$$k_0^0 > k_0^1 \textit{ or } k_1^0 \neq k_1^1 \textit{ or } k_0^1 \neq k_0^2 \textit{ or } k_1^1 < k_1^2$$

and

$$k_0^0 > k_0^2 \textit{ or } k_1^0 < k_1^2$$

Proof. The previous two theorems should provide enough intuition here that we can abandon our rigorous case-based approach. If we shift α by some $j > 0$ symbols to the right and it self-overlaps, the '01' pattern starting at bit k_0^0 must line up with either bit $k_0^0 + k_1^0 + k_0^1$ or bit $k_0^0 + k_1^0 + k_0^1 + k_1^1 + k_0^2$. In either case, the inequalities specified above guarantee that some block of 0's or 1's will not correctly line up with the corresponding block in the shift. □

Theorem 4.4. *The minterm* $\alpha = 0^{k_0^0}1^{k_1^0}0^{k_0^1}1^{k_1^1}...0^{k_0^j}1^{k_1^j}$ *is self-avoiding for* $k_0^0, k_1^0, k_0^1, k_1^1, ..., k_0^j, k_1^j > 0$ *s.t. for all* $0 \leq m < j$*:*

$$\Big(k_0^0 > k_0^{1+m} \textit{ or } k_1^{j-1-m} < k_1^j \textit{ or } k_0^i \neq k_0^{i+1+m} \textit{ for some } 1 \leq i \leq j-1-m$$
$$\textit{or } k_1^i \neq k_1^{i+1+m} \textit{ for some } 0 \leq i \leq j-2-m\Big)$$

Proof. This extends the previous three proofs in the logical way. Here, m represents the number of sets of blocks $0\cdots01\cdots1$ by which we shift α to try to align it with its shift. □

Fact 4.5. The minterm 00**1101, 00*1101, and 00*101 are self-avoiding.
Because these minterms contain a '*', Theorem (3.6) tells us that the associated functions MATCH_α have polynomial complexity strictly less than n for all input sizes n.

Proof. This is clear by enumeration. □

Fact 4.6. If α starts and ends with the same bit, then α is not self-avoiding.

Proof. We assume $\alpha[1] = \alpha[\text{size}(\alpha)]$. Then $\text{PREFIX}_\alpha(\text{s}_{\text{size}(\alpha)-1}(\alpha)) = 1$. □

Fact 4.7. If α begins with a '*', then α is not self-avoiding.

Proof. We assume $\alpha[1] = *$. Then $\text{PREFIX}_\alpha(\text{s}_{\text{size}(\alpha)-1}(\alpha)) = 1$. □

4.2 Minterms that Self-Overlap Only at the First Shift

Fact 4.8. The minterm $0^{k_0}*1^{k_1}$ for $k_0, k_1 > 0$ is not self-avoiding, but only self-overlaps at a single shift: $\text{PREFIX}_\alpha(\text{s}_1(\alpha)) = 1$.

Fact 4.9. 000*1**1 also self-overlaps at only s_1. Because this minterm contains '**', Theorem 3.9 tells us that the associated function MATCH_α has polynomial complexity strictly less than n for all input sizes n.

4.3 Minterms that Self-Overlap Only at the Second Shift

Fact 4.10. $010, 101, *10, *01, 0101, 1010, 0*01, 1*10, 01*1, 10*0$ all self-overlap at only the shift s_2: $\text{PREFIX}_\alpha(\text{s}_2(\alpha)) = 1$.

4.4 Minterms that Self-Overlap at Every Shift

Fact 4.11. Any monotone minterm $\alpha \in \{1, *\}^k$ self-overlaps at every shift. That is, $\forall\ i$ s.t. $0 < i < \text{size}(\alpha),\ \text{PREFIX}_\alpha(\text{s}_i(\alpha)) = 1$.

Chapter 5

Conclusions and Future Research

5.1 Polynomial Complexity

Based on our results at the end of Chapter 3 and our computational results in Appendix B, one feels intuitively that self-avoiding minterms are in some way 'building blocks' for all minterms. If we could take any minterm α and 'separate' it into disjoint cases of self-avoiding α_i s.t.

$$\text{MATCH}_\alpha = \text{OR}_i(\text{MATCH}_{\alpha_i})$$

$$\text{AND}_2(\text{MATCH}_{\alpha_i}, \text{MATCH}_{\alpha_j}) = 0 \ \forall\, i \neq j$$

then $p_{\text{MATCH}_\alpha} = \sum_i p_{\text{MATCH}_{\alpha_i}}$, and all coefficients of p_{MATCH_α} could be exactly determined by analysis of self-avoiding α_i. This would be a satisfying result, whose characterization rests on determining this separation of non-self-avoiding α into self-avoiding α_i.

5.2 Query Complexity

We proved in Chapter 2 that all monotone minterm-cyclic functions are evasive, and we proved the weaker result that all minterm-cyclic functions require n queries if we restrict ourselves to a fixed query ordering. (Recall: this corresponds to each level of the decision tree querying a specific bit.) It would be helpful to characterize the query complexity of MATCH_α in general for non-monotone α.

This evaluation of query complexity could proceed in one of several ways: First, one could try to directly prove the query complexity of such functions. That is, one could either prove that all minterm-cyclic functions are evasive, or characterize the cases in which they are not evasive. Secondly, one could search for the maximum difference between $\text{D}(f)$ and $\deg(f)$ for minterm-cyclic f. Note that we have not truly explored this direction, but we have not yet found any minterm-cyclic f s.t. $\deg(f) < \text{D}(f) - 1$, where the difference is strictly greater than 1.

Appendix A

Code

A.1 Macaulay2 Code to Generate Coefficients

First, we generate a list of minterms $\alpha = (A_0, A_1)$ and represent them in the following text form, one on each line:

```
{0}.{1,3}
```

Given a text file in this format, the following Macaulay2 code generates the degree-n coefficients of MATCH_α for each minterm α, using Equation (1.3). This code completes the code created by Talmage in [10] by adding the parsing of the non-monotone case. A '0' in the output represents a non-saturated function. (See Chapter 1 for more detail.)

mincyc.m2

```
matchpoly = (n, zeros, ones) -> (
        R = ZZ[x_0 .. x_n] / ((0..<n) / (i -> x_i^2 - x_i));
        1 - product(n, i -> 1-
        product(length ones, j -> x_((i + ones_j) % n)) *
        product(length zeros, j -> 1 - x_((i + zeros_j) % n))
        )
)

maxcoeff = (n, zeros, ones) -> (
        p = matchpoly(n, zeros, ones);
        coefficient(product(n, i -> x_i), p)
)

printToFile = (zeros, ones, outputFile, maxN) -> (
        outputFile << zeros << ones << "; ";
        scan(1..maxN, (n -> outputFile << maxcoeff(n, zeros, ones) | "; "));
        outputFile << endl;
        outputFile << flush;
)

mintermsFromFile = (mintermFile, outputFile, maxN) -> (
        minterms = lines get mintermFile;
```

```
        scan(minterms, (minterm ->
                printToFile({}, value(minterm), outputFile, maxN)
                )
        );
)

autoMinterms = (mintermFile, coefficientFile, k, maxN) -> (
        outputFile = coefficientFile << ";";
        scan(maxN, n -> outputFile << n+1 << "; ");
        outputFile << endl;
        for i from 1 to k do (
                minterms = createMintermFile(i, mintermFile);
                mintermsFromFile(mintermFile, outputFile, maxN);
        );
        outputFile << close;
)

mintermsFromNmFile = (mintermFile, outputFile, maxN) -> (
        minterms = lines get mintermFile;
        scan(minterms, (line ->
                printToFile(value(first(separate(".", line))),
                            value(last(separate(".", line))),
                            outputFile,
                            maxN)
                )
        );
        outputFile << flush;
        outputFile << close;
)
```

A.2 C Code to Generate Overlaps

We suggest that characterization of the 'overlaps' of a minterm α help in computing the coefficients of MATCH_α. Specifically, Theorem 3.6 gives the exact coefficients in the case of self-avoiding α. The following C code generates a characterization of the intersecting overlaps for any minterm α. A '0' in the output position j indicates that $\text{PREFIX}_\alpha(\text{cs}_j(\alpha)) = 0$, and a '1' in the output position j indicates that $\text{PREFIX}_\alpha(\text{cs}_j(\alpha)) = 1$. Thus, a row of all '0's represents a self-avoiding minterm.

overlaps.c

```
// @author Matt Harding
// Advised by Amit Chakrabarti
// Updated May 22, 2013
//
// Description:
// Creates a text file of minterms
//
// Usage:
//   ./overlaps max_size filename

#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define BIT(i,x)              ( ( (x) >> i ) & 1 )
#define M_IS_VALID(m0,m1)     ( ( m0 & m1 ) == 0 )
#define POW(i)                (((unsigned long long)1)<<i)
#define VAL(x)                ( POW(x)-1 )

int max_index(int x);
char* minterm_to_string(int m0, int m1);
char* minterm_to_string_2(int m0, int m1);
int overlap(int m0, int m1, int max, int i);

int main(int argc, char* argv[]) {

  /* Open log file */
  FILE *fp;
  if (argc > 2) {
    fp = fopen(argv[2], "w");
  } else if (argc < 2) {
    printf("\n\nUsage: ./createmintermfile max_size [filename]\n\n");
    return -1;
  }

  int max_size = atoi(argv[1]);
  if (max_size <= 0) {
    printf("\n\nUsage: ./createmintermfile max_size [filename]\n\n");
    return -1;
  }

  int val_max_size = VAL(max_size);
```

```
  // Cycle through all minterms in a double for-loop
  int m0, m1;
  for (m0 = 0; m0 <= val_max_size; m0++) {
    for (m1 = 0; m1 <= val_max_size; m1++) {
      if ((m0 == 0) && (m1 == 0))
        continue; // Skip the trivial minterm

      if (!M_IS_VALID(m0,m1) || !BIT(0, m0|m1))
        continue; // Skip invalid minterm or minterm with leading *s

      // Print minterm
      char* str = minterm_to_string(m0,m1);
      if (argc > 2) {
        fprintf(fp, "\n%s:", str);
      } else {
        printf("\n%s : ", str);
      }
      free(str);

      // Determine and print its overlaps
      int i;
      int max0 = max_index(m0);
      int max1 = max_index(m1);
      int max = (max0 > max1) ? (max0) : (max1);
      for (i=1; i<=max; i++) {
        if (argc > 2) {
          fprintf(fp, "%d,", overlap(m0, m1, max, i));
        } else {
          printf("%d,", overlap(m0, m1, max, i));
        }

      }
    }
  }

  // Finish output
  if (argc > 2) {
    fprintf(fp, "\n");
    fclose(fp);
  } else {
    printf("\n");
  }
  return 0;
}

/* Return the highest-index 1 bit in binary string represented by x */
int max_index(int x) {
  int i=0;
  while (x>>i != 0) {
    i++;
  }
  return i-1;
}
```

```
// Represent minterm {m0, m1} in the form 10*010
char* minterm_to_string (int m0, int m1) {
  int i;
  int max0 = max_index(m0);
  int max1 = max_index(m1);
  int max = (max0 > max1) ? (max0) : (max1);
  char* retval = calloc(max+1, sizeof(char));
  for (i=0; i<=max; i++) {
    if (BIT(i,m0))
      strcat(retval, "0");
    else if (BIT(i,m1))
      strcat(retval, "1");
    else
      strcat(retval, "*");
  }
  return retval;
}

// Represent minterm {m0, m1} in the form {1, 3, 5}, {0, 4}
char* minterm_to_string_2 (int m0, int m1) {
  int i;
  int max0 = max_index(m0);
  int max1 = max_index(m1);
  int max = (max0 > max1) ? (max0) : (max1);
  int added_char = 0;
  char* retval = calloc(6*(max+1) +2, sizeof(char));
  char* intstring = calloc(max+2, sizeof(char));
  strcat(retval, "{");
  for (i=0; i<=max; i++) {
    if (BIT(i,m0)) {
      if (added_char == 0)
        added_char = 1;
      else
        strcat(retval, ",");
      sprintf(intstring, "%d", i);
      strncat(retval, intstring, 2);
    }
  }
  strncat(retval, "}.{", 3);
  added_char = 0;
  for (i=0; i<=max; i++) {
    if (BIT(i,m1)) {
      if (added_char == 0)
        added_char = 1;
      else
        strcat(retval, ",");
      sprintf(intstring, "%d", i);
      strncat(retval, intstring, 2);
    }
  }
  strcat(retval, "}");

  return retval;
```

```
}

// Return 1 iff this minterm self-overlaps at i
int overlap(int m0, int m1, int max, int i) {
  int j;
  for (j=i; j<=max; j++) {
    if ((BIT(j, m0) && BIT(j-i, m1)) || (BIT(j, m1) && BIT(j-i, m0))) {
      return 0;   // Bit j conflicts with bit j-i
    }
  }
  return 1;       // No conflicts
}
```

Appendix B

Degree-n Coefficients of $\textsc{Match}_\alpha$ for Monotone α

The following table contains the data generated by `autoMinterms` from Appendix A.1. Each row represents a single monotone minterm-cyclic function $\textsc{Match}_\alpha$, given by the minterm $\alpha \in \{1, *\}^k$ in the leftmost column. Each column represents an input size n. In each cell is the coefficient of the degree-n term in the unique multilinear polynomial representing $\textsc{Match}_\alpha$. This value is 0 if and only if the polynomial complexity of the function $\textsc{Match}_\alpha$ at input size n is strictly less than n. Thus, if a row (extended infinitely to the right) consists of all non-zero coefficients, then $\textsc{Match}_\alpha$ is *saturated*.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1
11	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-2
1*1	1	1	-2	-1	1	-4	1	-1	-2	-1	1	-4	1	-1	-2	-1	1	-4	1	-1	-2	-1	1	-4	1	-1	-2	-1	1	-4
111	1	1	1	-3	1	1	1	-3	1	1	1	-3	1	1	1	-3	1	1	1	-3	1	1	1	-3	1	1	1	-3	1	1
1**1	1	1	1	1	1	1	1	1	-8	1	1	1	1	1	1	1	1	-8	1	1	1	1	1	1	1	1	-8	1	1	1
11*1	1	1	1	-3	1	4	-6	-3	7	1	-10	0	14	-6	-17	13	18	-23	-18	37	12	-54	1	72	-24	-90	61	102	-115	-101
1*11	1	1	1	-3	1	4	-6	-3	7	1	-10	0	14	-6	-17	13	18	-23	-18	37	12	-54	1	72	-24	-90	61	102	-115	-101
1111	1	1	1	1	-4	1	1	1	1	-4	1	1	1	1	-4	1	1	1	1	-4	1	1	1	1	-4	1	1	1	1	-4
1***1	1	1	1	1	1	-4	1	-1	-2	-1	1	-16	1	-1	-2	-1	1	-4	1	-1	-2	-1	1	-16	1	-1	-2	-1	1	-4
11**1	1	1	1	1	1	4	1	-7	7	1	-10	4	1	1	-2	-7	18	-5	-18	21	-2	-10	1	-4	26	-25	-20	57	-28	-26
1*1*1	1	1	1	1	1	-1	1	-9	1	-1	1	-1	1	-1	1	-9	1	-1	1	-1	1	-1	1	-9	1	-1	1	-1	1	-1
111*1	1	1	1	1	-4	1	8	-3	-8	-4	12	9	-12	-20	11	29	1	-44	-18	52	50	-54	-91	33	146	14	-197	-108	233	251
1**11	1	1	1	1	1	4	1	-7	7	1	-10	4	1	1	-2	-7	18	-5	-18	21	-2	-10	1	-4	26	-25	-20	57	-28	-26
11*11	1	1	1	1	-4	-2	1	-3	-11	-4	1	-6	-12	-13	-7	-3	-16	-29	-18	-8	-23	-43	-45	-30	-29	-64	-92	-73	-57	-97
1*111	1	1	1	1	-4	1	8	-3	-8	-4	12	9	-12	-20	11	29	1	-44	-18	52	50	-54	-91	33	146	14	-197	-108	233	251
11111	1	1	1	1	1	-5	1	1	1	1	1	-5	1	1	1	1	1	-5	1	1	1	1	1	-5	1	1	1	1	1	-5
1****1	1	1	1	1	1	-2	1	1	-2	1	1	-2	1	1	-32	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-32
11***1	1	1	1	1	1	1	-6	-7	1	11	-10	1	14	-6	-14	-7	18	1	-18	11	36	-32	-45	41	26	-38	-26	78	30	-124
1*1**1	1	1	1	1	1	4	1	-3	7	11	-10	0	14	1	-17	-3	18	-5	-37	7	40	-32	-45	48	51	-64	-47	109	59	-151
111**1	1	1	1	1	1	1	8	-3	-8	1	12	-3	-12	-6	16	13	-16	-26	20	37	-13	-54	1	69	26	-90	-62	102	117	-104
1**1*1	1	1	1	1	1	4	1	-3	7	11	-10	0	14	1	-17	-3	18	-5	-37	7	40	-32	-45	48	51	-64	-47	109	59	-151
11*1*1	1	1	1	1	1	1	1	-3	-8	1	12	-3	-12	1	16	-3	-16	-8	20	17	-20	-32	24	45	-24	-64	19	81	1	-104
1*11*1	1	1	1	1	1	-2	1	1	-11	1	1	-2	1	1	-2	1	1	-11	1	1	-2	1	1	-2	1	1	-11	1	1	-2
1111*1	1	1	1	1	1	-5	1	9	1	-4	-10	-5	14	15	-4	-23	-16	13	39	16	-41	-54	1	75	71	-38	-134	-69	117	200
1***11	1	1	1	1	1	1	-6	-7	1	11	-10	1	14	-6	-14	-7	18	1	-18	11	36	-32	-45	41	26	-38	-26	78	30	-124
11**11	1	1	1	1	1	1	1	1	1	1	1	1	-12	1	1	1	1	1	1	1	1	1	1	1	1	-12	1	1	1	1
1*1*11	1	1	1	1	1	1	1	-3	-8	1	12	-3	-12	1	16	-3	-16	-8	20	17	-20	-32	24	45	-24	-64	19	81	1	-104
111*11	1	1	1	1	1	-5	1	5	1	-14	-10	3	14	1	-19	-27	1	31	20	-38	-62	-10	70	59	-54	-142	-53	137	175	-65
1**111	1	1	1	1	1	1	8	-3	-8	1	12	-3	-12	-6	16	13	-16	-26	20	37	-13	-54	1	69	26	-90	-62	102	117	-104
11*111	1	1	1	1	1	-5	1	5	1	-14	-10	3	14	1	-19	-27	1	31	20	-38	-62	-10	70	59	-54	-142	-53	137	175	-65
1*1111	1	1	1	1	1	-5	1	9	1	-4	-10	-5	14	15	-4	-23	-16	13	39	16	-41	-54	1	75	71	-38	-134	-69	117	200
111111	1	1	1	1	1	1	-6	1	1	1	1	1	1	-6	1	1	1	1	1	1	-6	1	1	1	1	1	1	-6	1	1
1*****1	1	1	1	1	1	1	1	-1	-8	-1	1	-1	1	-1	1	-1	1	-64	1	-1	1	-1	1	-1	1	-1	-8	-1	1	-1
11****1	1	1	1	1	1	1	1	-3	7	11	1	-18	14	1	-17	13	18	-29	-18	7	-2	1	1	-18	26	-38	-47	109	30	-127
1*1***1	1	1	1	1	1	1	-6	-9	7	-1	-10	-16	14	-36	-17	-9	18	-49	-18	-1	12	-100	1	0	-24	-196	61	-36	-115	-289
111***1	1	1	1	1	1	1	1	-3	-8	1	12	-3	-12	1	16	-3	-16	-8	20	17	-20	-32	24	45	-24	-64	19	81	1	-104
1**1**1	1	1	1	1	1	1	1	-3	1	1	1	-27	1	1	1	-3	1	1	1	-3	1	1	1	-27	1	1	1	-3	1	1
11*1**1	1	1	1	1	1	1	8	1	-2	1	12	-8	-12	-6	-2	1	-16	-14	20	21	5	-10	24	40	1	-38	-29	22	1	-86
1*11**1	1	1	1	1	1	1	1	-3	-2	-4	12	-12	-12	1	-7	-3	-16	-14	1	-8	-2	-32	1	12	-29	-12	-29	-3	1	-61
1111**1	1	1	1	1	1	1	1	9	1	-14	1	1	14	1	-19	-7	1	37	1	-34	-20	1	70	9	-54	-64	1	141	30	-89
1***1*1	1	1	1	1	1	1	-6	-9	7	-1	-10	-16	14	-36	-17	-9	18	-49	-18	-1	12	-100	1	0	-24	-196	61	-36	-115	-289

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
11**1*1	1	1	1	1	1	1	8	-3	-11	1	12	-12	-38	22	28	-19	-33	13	58	17	-100	-54	139	84	-224	-168	313	242	-347	-386
1*1*1*1	1	1	1	1	1	1	1	-1	1	-16	1	-1	1	-1	-4	-1	1	-1	1	-16	1	-1	1	-1	-4	-1	1	-1	1	-16
111*1*1	1	1	1	1	1	1	1	-3	1	-4	1	-3	1	1	-4	-3	1	1	1	-8	1	1	1	-3	-4	1	1	-3	1	-4
1**11*1	1	1	1	1	1	1	1	-3	-2	-4	12	-12	-12	1	-7	-3	-16	-14	1	-8	-2	-32	1	12	-29	-12	-29	-3	1	-61
11*11*1	1	1	1	1	1	1	1	1	7	-14	-10	10	1	1	-7	-15	1	7	20	-14	-44	12	24	10	-4	-51	-20	29	59	13
1*111*1	1	1	1	1	1	1	1	-3	1	-9	1	-3	1	1	1	-3	1	1	1	-13	1	1	1	-3	1	1	1	-3	1	-9
11111*1	1	1	1	1	1	1	-6	1	10	1	1	-5	-12	-6	16	17	1	-14	-18	-19	15	45	24	-29	-49	-38	10	78	88	-20
1****11	1	1	1	1	1	1	1	-3	7	11	1	-18	14	1	-17	13	18	-29	-18	7	-2	1	1	-18	26	-38	-47	109	30	-127
11***11	1	1	1	1	1	1	1	-3	1	1	1	-3	1	-13	1	-3	1	1	1	-3	1	1	1	-3	1	1	1	-17	1	1
1*1**11	1	1	1	1	1	1	8	-3	-11	1	12	-12	-38	22	28	-19	-33	13	58	17	-100	-54	139	84	-224	-168	313	242	-347	-386
111**11	1	1	1	1	1	1	1	5	1	1	-10	9	1	1	-14	5	1	1	1	-3	1	-10	24	-7	1	-25	28	-3	1	-14
1**1*11	1	1	1	1	1	1	8	1	-2	1	12	-8	-12	-6	-2	1	-16	-14	20	21	5	-10	24	40	1	-38	-29	22	1	-86
11*1*11	1	1	1	1	1	1	1	1	1	-9	1	25	27	-13	-29	-15	1	1	1	-9	1	45	47	-71	-149	-25	163	155	1	-99
1*11*11	1	1	1	1	1	1	1	1	7	-14	-10	10	1	1	-7	-15	1	7	20	-14	-44	12	24	10	-4	-51	-20	29	59	13
1111*11	1	1	1	1	1	1	-6	1	10	6	-10	-17	-12	8	26	17	-16	-50	-37	26	78	56	-45	-137	-104	66	226	176	-115	-390
1***111	1	1	1	1	1	1	1	-3	-8	1	12	-3	-12	1	16	-3	-16	-8	20	17	-20	-32	24	45	-24	-64	19	81	1	-104
11**111	1	1	1	1	1	1	1	5	1	1	-10	9	1	1	-14	5	1	1	1	-3	1	-10	24	-7	1	-25	28	-3	1	-14
1*1*111	1	1	1	1	1	1	1	-3	1	-4	1	-3	1	1	-4	-3	1	1	1	-8	1	1	1	-3	-4	1	1	-3	1	-4
111*111	1	1	1	1	1	1	-6	-3	1	-4	-10	-21	-12	-6	-4	-19	-33	-41	-37	-28	-27	-54	-91	-117	-104	-90	-107	-178	-260	-310
1**1111	1	1	1	1	1	1	1	9	1	-14	1	1	14	1	-19	-7	1	37	1	-34	-20	1	70	9	-54	-64	1	141	30	-89
11*1111	1	1	1	1	1	1	-6	1	10	6	-10	-17	-12	8	26	17	-16	-50	-37	26	78	56	-45	-137	-104	66	226	176	-115	-390
1*11111	1	1	1	1	1	1	-6	1	10	1	1	-5	-12	-6	16	17	1	-14	-18	-19	15	45	24	-29	-49	-38	10	78	88	-20
1111111	1	1	1	1	1	1	1	-7	1	1	1	1	1	1	1	-7	1	1	1	1	1	1	1	-7	1	1	1	1	1	1
1******1	1	1	1	1	1	1	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-128	1	1	-2	1	1	-2	1	1	-2
11*****1	1	1	1	1	1	1	1	-3	7	1	-10	-18	1	29	-17	-3	18	-29	-18	37	-2	-10	-45	-18	26	1	-20	25	30	-107
1*1****1	1	1	1	1	1	1	1	-3	1	11	-10	-3	1	29	-14	13	18	1	-18	7	1	-10	1	-3	51	-25	-26	25	59	-124
111****1	1	1	1	1	1	1	1	1	-8	1	12	1	-12	1	16	1	-16	-8	1	21	1	-32	1	49	1	-64	-8	57	30	-44
1**1***1	1	1	1	1	1	1	1	-7	7	1	1	4	14	29	-17	-7	18	-23	-18	1	-2	1	-45	-4	26	-64	-47	29	59	-11
11*1***1	1	1	1	1	1	1	1	-3	-11	-4	12	-12	-12	8	8	-19	-50	13	20	-28	-51	-10	70	-36	-129	-38	97	32	-231	-106
1*11***1	1	1	1	1	1	1	1	-3	-8	1	12	-3	-38	8	31	-19	-50	10	58	-23	-90	-32	93	69	-149	-168	181	228	-289	-419
1111***1	1	1	1	1	1	1	1	1	1	-14	1	13	1	1	-19	1	18	1	1	-34	1	45	1	-11	-54	1	82	1	-28	-89
1***1**1	1	1	1	1	1	1	1	-7	7	1	1	4	14	29	-17	-7	18	-23	-18	1	-2	1	-45	-4	26	-64	-47	29	59	-11
11**1**1	1	1	1	1	1	1	1	-3	-2	-4	12	-12	-12	1	-7	13	-16	-14	20	-8	19	-32	1	36	-29	14	-56	25	59	-91
1*1*1**1	1	1	1	1	1	1	1	-3	1	1	12	9	-12	1	16	-3	-16	1	20	-3	-20	-10	1	9	1	-38	1	53	1	-74
111*1**1	1	1	1	1	1	1	1	9	1	-14	-10	25	14	-6	-49	-23	52	55	-18	-114	-55	144	185	-63	-304	-168	379	498	-173	-869
1**11**1	1	1	1	1	1	1	1	1	-11	-4	1	-2	-12	1	-7	1	1	-11	1	-4	-2	1	1	-2	-4	-12	-11	1	1	-7
11*11**1	1	1	1	1	1	1	1	5	7	1	-10	18	14	1	-2	5	1	7	1	-3	-23	-32	24	2	-49	-38	7	-3	1	-2
1*111**1	1	1	1	1	1	1	1	5	1	-14	1	15	14	-20	-19	-11	35	13	-18	-78	8	67	70	-97	-104	-12	217	88	-144	-335
11111**1	1	1	1	1	1	1	1	1	10	1	-10	-5	1	1	16	1	-16	-14	1	21	22	-10	-22	-29	1	53	37	-27	-57	-50
1****1*1	1	1	1	1	1	1	1	-3	1	11	-10	-3	1	29	-14	13	18	1	-18	7	1	-10	1	-3	51	-25	-26	25	59	-124
11***1*1	1	1	1	1	1	1	1	-3	-8	1	1	-3	-12	8	31	-19	-33	-8	20	-3	-27	1	24	-3	-49	-38	46	60	-86	-149

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1*1**1*1	1	1	1	1	1	1	1	1	1	1	1	1	-12	1	1	1	-16	1	1	1	1	1	1	1	1	-12	1	1	1	1
111**1*1	1	1	1	1	1	1	1	9	1	-9	-10	13	14	-20	-44	-7	18	37	1	-49	-13	100	116	-27	-174	-90	163	176	-144	-414
1**1*1*1	1	1	1	1	1	1	1	-3	1	1	12	9	-12	1	16	-3	-16	1	20	-3	-20	-10	1	9	1	-38	1	53	1	-74
11*1*1*1	1	1	1	1	1	1	1	-3	1	-4	-10	-3	14	1	-19	-3	18	1	-18	-8	22	-10	-22	21	21	-38	-26	53	30	-79
1*11*1*1	1	1	1	1	1	1	1	1	1	1	-10	-5	1	1	1	1	1	-5	1	1	1	-10	1	-5	1	1	1	1	1	-5
1111*1*1	1	1	1	1	1	1	1	1	1	6	-10	-5	1	1	11	1	-16	-5	1	6	22	-10	-22	-5	-4	27	28	-27	-28	-15
1***11*1	1	1	1	1	1	1	1	-3	-8	1	12	-3	-38	8	31	-19	-50	10	58	-23	-90	-32	93	69	-149	-168	181	228	-289	-419
11**11*1	1	1	1	1	1	1	1	5	1	-4	-10	21	14	-27	-34	21	35	-17	-37	-28	22	34	47	-67	-129	66	244	-3	-347	-94
1*1*11*1	1	1	1	1	1	1	1	1	1	1	-10	-5	1	1	1	1	1	-5	1	1	1	-10	1	-5	1	1	1	1	1	-5
111*11*1	1	1	1	1	1	1	1	1	10	6	-21	-17	1	29	26	-15	-50	-32	20	86	64	-65	-160	-89	146	287	91	-307	-463	-90
1**111*1	1	1	1	1	1	1	1	5	1	-14	1	15	14	-20	-19	-11	35	13	-18	-78	8	67	70	-97	-104	-12	217	88	-144	-335
11*111*1	1	1	1	1	1	1	1	-3	10	1	-10	-21	14	1	16	-19	1	-32	20	-3	64	-54	-22	-69	101	-12	91	-143	30	-170
1*1111*1	1	1	1	1	1	1	1	1	1	-4	-10	1	1	1	-4	1	1	1	1	-4	1	-10	1	1	-4	1	1	1	1	-4
111111*1	1	1	1	1	1	1	1	-7	1	11	1	1	1	-6	-14	-7	18	19	1	-9	-6	-21	-22	17	51	27	-26	-34	-28	-34
1*****11	1	1	1	1	1	1	1	-3	7	1	-10	-18	1	29	-17	-3	18	-29	-18	37	-2	-10	-45	-18	26	1	-20	25	30	-107
11****11	1	1	1	1	1	1	1	1	-11	1	1	-2	1	1	-2	1	-16	-47	-18	1	-2	1	1	-2	1	1	-11	1	1	-2
1*1***11	1	1	1	1	1	1	1	-3	-8	1	1	-3	-12	8	31	-19	-33	-8	20	-3	-27	1	24	-3	-49	-38	46	60	-86	-149
111***11	1	1	1	1	1	1	1	1	1	1	1	-11	14	1	1	-15	1	1	1	1	1	1	1	-11	26	-12	1	-27	30	1
1**1**11	1	1	1	1	1	1	1	-3	-2	-4	12	-12	-12	1	-7	13	-16	-14	20	-8	19	-32	1	36	-29	14	-56	25	59	-91
11*1**11	1	1	1	1	1	1	1	9	7	-4	1	4	1	-20	-37	-7	18	-5	-18	-24	26	67	47	-12	-54	27	88	8	-144	-181
1*11**11	1	1	1	1	1	1	1	5	1	-4	-10	21	14	-27	-34	21	35	-17	-37	-28	22	34	47	-67	-129	66	244	-3	-347	-94
1111**11	1	1	1	1	1	1	1	1	10	-4	1	-11	1	15	-4	1	-16	-8	20	-4	22	-21	-22	13	-4	53	-17	-41	1	-34
1***1*11	1	1	1	1	1	1	1	-3	-11	-4	12	-12	-12	8	8	-19	-50	13	20	-28	-51	-10	70	-36	-129	-38	97	32	-231	-106
11**1*11	1	1	1	1	1	1	1	9	7	-4	1	4	1	-20	-37	-7	18	-5	-18	-24	26	67	47	-12	-54	27	88	8	-144	-181
1*1*1*11	1	1	1	1	1	1	1	-3	1	-4	-10	-3	14	1	-19	-3	18	1	-18	-8	22	-10	-22	21	21	-38	-26	53	30	-79
111*1*11	1	1	1	1	1	1	1	1	10	6	-10	-11	1	43	11	-31	-33	-8	39	26	1	-32	-91	-35	121	157	-17	-293	-173	231
1**11*11	1	1	1	1	1	1	1	5	7	1	-10	18	14	1	-2	5	1	7	1	-3	-23	-32	24	2	-49	-38	7	-3	1	-2
11*11*11	1	1	1	1	1	1	1	1	-2	1	-10	-26	-12	1	-2	1	1	-2	1	1	-2	-10	-45	-74	-49	-12	-2	1	1	-2
1*111*11	1	1	1	1	1	1	1	-3	10	1	-10	-21	14	1	16	-19	1	-32	20	-3	64	-54	-22	-69	101	-12	91	-143	30	-170
11111*11	1	1	1	1	1	1	1	-7	1	11	12	-5	-12	-20	-14	9	35	31	1	-49	-69	-32	47	107	101	-12	-161	-216	-86	170
1****111	1	1	1	1	1	1	1	1	-8	1	12	1	-12	1	16	1	-16	-8	1	21	1	-32	1	49	1	-64	-8	57	30	-44
11***111	1	1	1	1	1	1	1	1	1	1	1	-11	14	1	1	-15	1	1	1	1	1	1	1	-11	26	-12	1	-27	30	1
1*1**111	1	1	1	1	1	1	1	9	1	-9	-10	13	14	-20	-44	-7	18	37	1	-49	-13	100	116	-27	-174	-90	163	176	-144	-414
111**111	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-16	1	1	1	1	1	1	1	1	1	1	1	1	1
1**1*111	1	1	1	1	1	1	1	9	1	-14	-10	25	14	-6	-49	-23	52	55	-18	-114	-55	144	185	-63	-304	-168	379	498	-173	-869
11*1*111	1	1	1	1	1	1	1	1	10	6	-10	-11	1	43	11	-31	-33	-8	39	26	1	-32	-91	-35	121	157	-17	-293	-173	231
1*11*111	1	1	1	1	1	1	1	1	10	6	-21	-17	1	29	26	-15	-50	-32	20	86	64	-65	-160	-89	146	287	91	-307	-463	-90
1111*111	1	1	1	1	1	1	1	-7	1	6	1	-5	-25	-20	-4	9	18	-5	-37	-54	-48	1	47	35	-29	-129	-161	-76	59	150
1***1111	1	1	1	1	1	1	1	1	1	-14	1	13	1	1	-19	1	18	1	1	-34	1	45	1	-11	-54	1	82	1	-28	-89
11**1111	1	1	1	1	1	1	1	1	10	-4	1	-11	1	15	-4	1	-16	-8	20	-4	22	-21	-22	13	-4	53	-17	-41	1	-34
1*1*1111	1	1	1	1	1	1	1	1	1	6	-10	-5	1	1	11	1	-16	-5	1	6	22	-10	-22	-5	-4	27	28	-27	-28	-15

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
111*1111	1	1	1	1	1	1	1	-7	1	6	1	-5	-25	-20	-4	9	18	-5	-37	-54	-48	1	47	35	-29	-129	-161	-76	59	150
1**11111	1	1	1	1	1	1	1	1	10	1	-10	-5	1	1	16	1	-16	-14	1	21	22	-10	-22	-29	1	53	37	-27	-57	-50
11*11111	1	1	1	1	1	1	1	-7	1	11	12	-5	-12	-20	-14	9	35	31	1	-49	-69	-32	47	107	101	-12	-161	-216	-86	170
1*111111	1	1	1	1	1	1	1	-7	1	11	1	1	1	-6	-14	-7	18	19	1	-9	-6	-21	-22	17	51	27	-26	-34	-28	-34
11111111	1	1	1	1	1	1	1	1	-8	1	1	1	1	1	1	1	1	-8	1	1	1	1	1	1	1	1	-8	1	1	1
1*******1	1	1	1	1	1	1	1	1	-2	-1	1	-16	1	-1	-2	-1	1	-4	1	-1	-2	-1	1	-256	1	-1	-2	-1	1	-4
11******1	1	1	1	1	1	1	1	1	1	1	-10	1	14	29	1	-47	18	1	-37	21	1	-54	1	73	26	-64	-26	29	-28	1
1*1*****1	1	1	1	1	1	1	1	1	7	-1	-10	-16	1	-1	-2	-49	18	-49	-18	-1	-2	-100	1	-16	26	-1	-20	-1	-28	-4
111*****1	1	1	1	1	1	1	1	1	1	-4	12	-3	-12	1	11	-3	-16	1	20	-8	-20	-32	24	45	-29	-64	28	81	-28	-109
1**1****1	1	1	1	1	1	1	1	1	7	11	1	4	1	-6	-17	-47	18	-23	-18	11	12	1	1	76	51	1	-20	78	-28	-151
11*1****1	1	1	1	1	1	1	1	1	-8	1	12	-3	-12	8	16	-3	-50	10	39	17	-48	-10	70	21	-74	-90	100	144	-86	-194
1*11****1	1	1	1	1	1	1	1	1	-11	1	12	-12	-12	1	28	-19	-33	13	58	-3	-86	-10	93	12	-149	-90	205	137	-231	-266
1111****1	1	1	1	1	1	1	1	1	1	-4	1	13	1	-13	-4	1	18	1	-18	-4	1	23	1	-35	-4	27	28	-13	-57	-4
1***1***1	1	1	1	1	1	1	1	1	1	-1	1	-1	1	-1	1	-81	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1
11**1***1	1	1	1	1	1	1	1	1	-8	-4	1	1	-12	1	11	-23	-16	-8	20	-4	1	1	24	17	-29	-38	-8	1	-28	-19
1*1*1***1	1	1	1	1	1	1	1	1	-8	-16	12	-1	-12	-64	11	-9	1	-64	-18	-16	50	-144	-91	-81	146	-144	-197	-400	233	-121
111*1***1	1	1	1	1	1	1	1	1	1	-9	1	-3	14	-20	-14	-3	35	1	-18	-33	50	23	1	-75	1	14	82	-52	-57	-84
1**11***1	1	1	1	1	1	1	1	1	1	1	12	1	-12	1	16	-23	-16	1	20	1	-20	-10	1	17	1	-12	28	1	-28	-44
11*11***1	1	1	1	1	1	1	1	1	1	-14	-10	-9	14	-27	-19	-19	1	-23	-18	-58	-20	-54	1	-81	-79	-142	1	-115	-115	-275
1*111***1	1	1	1	1	1	1	1	1	1	-4	-10	-3	14	-13	-19	-3	18	19	-37	-48	22	34	1	-51	-54	14	55	11	-57	-109
11111***1	1	1	1	1	1	1	1	1	1	1	-10	-5	14	1	1	1	-16	-5	20	1	1	-10	-22	19	26	-12	1	-27	-28	55
1****1**1	1	1	1	1	1	1	1	1	7	11	1	4	1	-6	-17	-47	18	-23	-18	11	12	1	1	76	51	1	-20	78	-28	-151
11***1**1	1	1	1	1	1	1	1	1	-8	1	12	1	-12	8	16	-39	-33	10	58	21	-27	12	24	-31	-124	-38	154	148	-86	-134
1*1**1**1	1	1	1	1	1	1	1	1	-2	1	1	-12	-12	-20	-2	-3	-16	-14	20	17	5	1	1	12	1	-64	-56	-24	-28	-56
111**1**1	1	1	1	1	1	1	1	1	1	-14	1	21	14	-20	-19	-3	69	1	-37	-58	50	89	24	-171	-79	92	271	-52	-347	-239
1**1*1**1	1	1	1	1	1	1	1	1	-11	1	1	-6	1	-13	-2	-3	1	-29	1	-3	-2	1	1	-6	1	1	-11	-17	1	-2
11*1*1**1	1	1	1	1	1	1	1	1	1	1	1	3	14	-27	-14	29	35	-23	1	-23	1	1	47	-21	-24	-38	109	-59	-115	10
1*11*1**1	1	1	1	1	1	1	1	1	7	1	1	10	14	1	-2	-15	1	7	1	1	-2	-21	-22	10	1	-38	7	29	1	-2
1111*1**1	1	1	1	1	1	1	1	1	10	-4	-10	-17	14	29	11	-23	-50	-14	58	76	1	-120	-114	39	221	170	-152	-391	-202	365
1***11**1	1	1	1	1	1	1	1	1	1	1	12	1	-12	1	16	-23	-16	1	20	1	-20	-10	1	17	1	-12	28	1	-28	-44
11**11**1	1	1	1	1	1	1	1	1	1	-4	1	-17	14	1	-19	1	1	-5	1	-4	1	1	1	-17	21	-12	-26	29	1	-25
1*1*11**1	1	1	1	1	1	1	1	1	1	-4	1	-3	14	-13	-19	-3	1	1	-18	-28	1	1	1	-27	-29	-12	28	11	-57	-49
111*11**1	1	1	1	1	1	1	1	1	10	1	1	-21	14	15	31	-27	-16	-32	39	17	43	-65	-45	-85	126	66	91	-241	-86	-95
1**111**1	1	1	1	1	1	1	1	1	1	-9	1	-9	1	-13	1	-3	1	-5	1	-13	1	1	1	-9	1	1	1	-17	1	-15
11*111**1	1	1	1	1	1	1	1	1	1	6	1	-3	14	15	-4	5	1	1	1	2	1	-21	-22	-19	21	-12	-53	-17	1	6
1*1111**1	1	1	1	1	1	1	1	1	10	6	-10	-17	14	8	26	-7	-33	-32	1	66	36	-32	-114	-57	71	144	118	-160	-260	-120
111111**1	1	1	1	1	1	1	1	1	1	11	1	-11	1	-6	1	1	18	1	-18	-9	-6	23	1	13	1	-25	-26	-6	59	11
1*****1*1	1	1	1	1	1	1	1	1	7	-1	-10	-16	1	-1	-2	-49	18	-49	-18	-1	-2	-100	1	-16	26	-1	-20	-1	-28	-4
11****1*1	1	1	1	1	1	1	1	1	-8	1	12	-3	-12	1	16	-19	-50	-8	58	17	-83	-10	93	21	-124	-64	154	137	-202	-284
1*1***1*1	1	1	1	1	1	1	1	1	-11	-16	1	-4	-12	-1	-7	-9	-16	-121	-18	-16	-23	-1	-45	-36	-29	-144	-92	-169	-57	-49
111***1*1	1	1	1	1	1	1	1	1	1	-4	-10	-15	14	15	-19	-19	18	19	-18	-68	1	34	-22	-39	46	40	-80	-45	117	41

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1**1**1*1	1	1	1	1	1	1	1	1	-2	1	1	-12	-12	-20	-2	-3	-16	-14	20	17	5	1	1	12	1	-64	-56	-24	-28	-56
11*1**1*1	1	1	1	1	1	1	1	1	1	1	-10	1	14	-20	-14	17	1	-17	1	21	8	-10	1	25	-24	-38	55	8	-86	16
1*11**1*1	1	1	1	1	1	1	1	1	7	-4	-10	-12	14	1	-37	-19	18	31	-18	-68	19	34	1	-12	-54	-38	-20	53	88	-151
1111**1*1	1	1	1	1	1	1	1	1	10	6	-10	-23	14	15	11	-23	-50	-8	20	66	22	-54	-91	-15	171	144	-17	-265	-231	141
1***1*1*1	1	1	1	1	1	1	1	1	-8	-16	12	-1	-12	-64	11	-9	1	-64	-18	-16	50	-144	-91	-81	146	-144	-197	-400	233	-121
11**1*1*1	1	1	1	1	1	1	1	1	1	-4	1	-3	14	-6	-19	13	52	-17	-18	-28	50	23	-22	-75	21	14	55	-38	-57	-109
1*1*1*1*1	1	1	1	1	1	1	1	1	1	-1	1	-25	1	-1	1	-1	1	-1	1	-1	1	-1	1	-25	1	-1	1	-1	1	-1
111*1*1*1	1	1	1	1	1	1	1	1	1	1	1	-15	1	1	1	-3	1	1	1	-3	1	1	1	-15	1	1	1	-3	1	1
1**11*1*1	1	1	1	1	1	1	1	1	1	-4	1	-3	14	-13	-19	-3	1	1	-18	-28	1	1	1	-27	-29	-12	28	11	-57	-49
11*11*1*1	1	1	1	1	1	1	1	1	10	-4	-10	-17	1	29	11	-31	-33	-14	39	56	1	-76	-91	7	146	131	-71	-279	-173	215
1*111*1*1	1	1	1	1	1	1	1	1	10	1	-10	-15	14	-6	-14	13	18	-26	-18	37	15	-54	1	57	-24	-90	64	102	-115	-104
11111*1*1	1	1	1	1	1	1	1	1	1	1	12	-5	-12	-6	1	1	18	13	-18	-19	-6	-10	24	43	1	-38	-26	-34	1	85
1****11*1	1	1	1	1	1	1	1	1	-11	1	12	-12	-12	1	28	-19	-33	13	58	-3	-86	-10	93	12	-149	-90	205	137	-231	-266
11***11*1	1	1	1	1	1	1	1	1	1	-9	1	9	1	-27	-44	13	18	1	-37	-13	22	45	-22	-63	-74	79	82	-59	-260	-84
1*1**11*1	1	1	1	1	1	1	1	1	7	-4	-10	-12	14	1	-37	-19	18	31	-18	-68	19	34	1	-12	-54	-38	-20	53	88	-151
111**11*1	1	1	1	1	1	1	1	1	10	6	-10	-27	1	29	11	-59	-50	28	58	22	-20	-32	-22	5	121	157	-98	-395	-173	441
1**1*11*1	1	1	1	1	1	1	1	1	7	1	1	10	14	1	-2	-15	1	7	1	1	-2	-21	-22	10	1	-38	7	29	1	-2
11*1*11*1	1	1	1	1	1	1	1	1	10	1	1	-17	-12	15	16	-15	1	4	1	1	1	1	1	-17	1	14	10	-41	-28	70
1*11*11*1	1	1	1	1	1	1	1	1	-2	-4	1	-26	1	-6	-7	1	-16	-2	-18	-4	-9	-21	1	-50	-4	-25	-29	-6	-57	-7
1111*11*1	1	1	1	1	1	1	1	1	1	6	1	-17	-12	-6	11	17	1	-23	-37	-14	15	45	24	-41	-79	-64	28	106	88	-45
1***111*1	1	1	1	1	1	1	1	1	1	-4	-10	-3	14	-13	-19	-3	18	19	-37	-48	22	34	1	-51	-54	14	55	11	-57	-109
11**111*1	1	1	1	1	1	1	1	1	1	6	-10	-21	1	29	-4	-27	-50	-5	20	22	-20	-54	-68	-37	96	105	-53	-255	-144	150
1*1*111*1	1	1	1	1	1	1	1	1	10	1	-10	-15	14	-6	-14	13	18	-26	-18	37	15	-54	1	57	-24	-90	64	102	-115	-104
111*111*1	1	1	1	1	1	1	1	1	1	1	12	-9	-25	-20	16	13	1	-5	-18	-43	-27	12	47	39	-24	-129	-107	32	117	70
1**1111*1	1	1	1	1	1	1	1	1	10	6	-10	-17	14	8	26	-7	-33	-32	1	66	36	-32	-114	-57	71	144	118	-160	-260	-120
11*1111*1	1	1	1	1	1	1	1	1	1	6	12	-23	-12	-6	11	17	1	1	-37	-14	-6	34	70	-23	-54	-90	1	78	88	81
1*11111*1	1	1	1	1	1	1	1	1	1	1	1	-17	1	1	1	1	1	-5	1	1	1	1	1	-17	1	1	1	1	1	-5
1111111*1	1	1	1	1	1	1	1	1	-8	1	12	1	1	1	1	-7	-16	-8	20	21	1	-10	1	-7	-24	-25	19	57	30	-29
1******11	1	1	1	1	1	1	1	1	1	1	-10	1	14	29	1	-47	18	1	-37	21	1	-54	1	73	26	-64	-26	29	-28	1
11*****11	1	1	1	1	1	1	1	1	1	-4	1	-3	1	1	-4	-3	1	1	1	-8	1	-43	1	-3	-4	1	1	-3	1	-4
1*1****11	1	1	1	1	1	1	1	1	-8	1	12	-3	-12	1	16	-19	-50	-8	58	17	-83	-10	93	21	-124	-64	154	137	-202	-284
111****11	1	1	1	1	1	1	1	1	1	-14	-10	-11	1	15	-4	-15	1	19	20	-14	-20	-32	-22	13	21	1	-26	-13	59	76
1**1***11	1	1	1	1	1	1	1	1	-8	1	12	1	-12	8	16	-39	-33	10	58	21	-27	12	24	-31	-124	-38	154	148	-86	-134
11*1***11	1	1	1	1	1	1	1	1	1	-14	-10	9	14	8	-19	13	18	1	-37	-58	-6	12	1	-39	-54	40	55	-80	-231	-179
1*11***11	1	1	1	1	1	1	1	1	1	-9	1	9	1	-27	-44	13	18	1	-37	-13	22	45	-22	-63	-74	79	82	-59	-260	-84
1111***11	1	1	1	1	1	1	1	1	1	6	1	1	-12	1	11	1	1	-17	1	6	1	1	1	1	-4	-12	1	29	1	-9
1***1**11	1	1	1	1	1	1	1	1	-8	-4	1	1	-12	1	11	-23	-16	-8	20	-4	1	1	24	17	-29	-38	-8	1	-28	-19
11**1**11	1	1	1	1	1	1	1	1	1	1	1	-3	1	-13	1	-3	1	1	1	-3	1	-21	1	-3	1	1	1	-17	1	1
1*1*1**11	1	1	1	1	1	1	1	1	1	-4	1	-3	14	-6	-19	13	52	-17	-18	-28	50	23	-22	-75	21	14	55	-38	-57	-109
111*1**11	1	1	1	1	1	1	1	1	10	1	-10	-11	14	22	1	-55	-67	-8	39	21	-34	-76	1	165	251	118	-152	-258	30	391
1**11**11	1	1	1	1	1	1	1	1	1	-4	1	-17	14	1	-19	1	1	-5	1	-4	1	1	1	-17	21	-12	-26	29	1	-25

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
11*11**11	1	1	1	1	1	1	1	1	10	1	1	3	14	15	16	-11	1	4	20	-3	-20	-43	-22	-13	1	-38	-71	-73	1	40
1*111**11	1	1	1	1	1	1	1	1	1	6	-10	-21	1	29	-4	-27	-50	-5	20	22	-20	-54	-68	-37	96	105	-53	-255	-144	150
11111**11	1	1	1	1	1	1	1	1	1	11	1	-5	-12	1	1	17	1	-5	-18	-9	1	23	24	-5	-24	-38	1	29	59	5
1****1*11	1	1	1	1	1	1	1	1	-8	1	12	-3	-12	8	16	-3	-50	10	39	17	-48	-10	70	21	-74	-90	100	144	-86	-194
11***1*11	1	1	1	1	1	1	1	1	1	-14	-10	9	14	8	-19	13	18	1	-37	-58	-6	12	1	-39	-54	40	55	-80	-231	-179
1*1**1*11	1	1	1	1	1	1	1	1	1	1	-10	1	14	-20	-14	17	1	-17	1	21	8	-10	1	25	-24	-38	55	8	-86	16
111**1*11	1	1	1	1	1	1	1	1	10	6	-10	-11	1	-6	11	-55	-33	-26	20	26	29	-10	24	69	121	131	-17	-146	-115	-99
1**1*1*11	1	1	1	1	1	1	1	1	1	1	1	3	14	-27	-14	29	35	-23	1	-23	1	1	47	-21	-24	-38	109	-59	-115	10
11*1*1*11	1	1	1	1	1	1	1	1	1	-4	-10	-21	1	29	-4	-67	-50	31	39	-28	-62	-54	-22	-21	-4	27	-53	-199	-86	230
1*11*1*11	1	1	1	1	1	1	1	1	10	1	1	-17	-12	15	16	-15	1	4	1	1	1	1	1	-17	1	14	10	-41	-28	70
1111*1*11	1	1	1	1	1	1	1	1	1	6	12	-5	-12	-13	11	33	18	-23	-37	-14	22	34	24	-5	-29	-64	-53	43	146	105
1***11*11	1	1	1	1	1	1	1	1	1	-14	-10	-9	14	-27	-19	-19	1	-23	-18	-58	-20	-54	1	-81	-79	-142	1	-115	-115	-275
11**11*11	1	1	1	1	1	1	1	1	10	1	1	3	14	15	16	-11	1	4	20	-3	-20	-43	-22	-13	1	-38	-71	-73	1	40
1*1*11*11	1	1	1	1	1	1	1	1	10	-4	-10	-17	1	29	11	-31	-33	-14	39	56	1	-76	-91	7	146	131	-71	-279	-173	215
111*11*11	1	1	1	1	1	1	1	1	1	1	1	-17	-25	-13	16	33	18	-23	-37	-19	22	45	24	-41	-99	-103	1	155	204	40
1**111*11	1	1	1	1	1	1	1	1	1	6	1	-3	14	15	-4	5	1	1	1	2	1	-21	-22	-19	21	-12	-53	-17	1	6
11*111*11	1	1	1	1	1	1	1	1	1	1	1	-15	-25	-13	1	-3	1	1	1	-3	1	1	1	-15	-49	-77	-53	-17	1	1
1*1111*11	1	1	1	1	1	1	1	1	1	6	12	-23	-12	-6	11	17	1	1	-37	-14	-6	34	70	-23	-54	-90	1	78	88	81
111111*11	1	1	1	1	1	1	1	1	-8	1	12	13	1	-6	-14	-23	-16	10	39	41	15	-32	-68	-67	-24	53	127	134	30	-134
1*****111	1	1	1	1	1	1	1	1	1	-4	12	-3	-12	1	11	-3	-16	1	20	-8	-20	-32	24	45	-29	-64	28	81	-28	-109
11****111	1	1	1	1	1	1	1	1	1	-14	-10	-11	1	15	-4	-15	1	19	20	-14	-20	-32	-22	13	21	1	-26	-13	59	76
1*1***111	1	1	1	1	1	1	1	1	1	-4	-10	-15	14	15	-19	-19	18	19	-18	-68	1	34	-22	-39	46	40	-80	-45	117	41
111***111	1	1	1	1	1	1	1	1	1	-4	1	1	1	1	-4	1	1	-17	1	-4	1	1	1	1	-4	1	1	1	1	-4
1**1**111	1	1	1	1	1	1	1	1	1	-14	1	21	14	-20	-19	-3	69	1	-37	-58	50	89	24	-171	-79	92	271	-52	-347	-239
11*1**111	1	1	1	1	1	1	1	1	10	6	-10	-11	1	-6	11	-55	-33	-26	20	26	29	-10	24	69	121	131	-17	-146	-115	-99
1*11**111	1	1	1	1	1	1	1	1	10	6	-10	-27	1	29	11	-59	-50	28	58	22	-20	-32	-22	5	121	157	-98	-395	-173	441
1111**111	1	1	1	1	1	1	1	1	1	6	1	1	1	-13	11	1	1	1	-18	6	1	1	1	1	-4	1	1	-13	30	-9
1***1*111	1	1	1	1	1	1	1	1	1	-9	1	-3	14	-20	-14	-3	35	1	-18	-33	50	23	1	-75	1	14	82	-52	-57	-84
11**1*111	1	1	1	1	1	1	1	1	10	1	-10	-11	14	22	1	-55	-67	-8	39	21	-34	-76	1	165	251	118	-152	-258	30	391
1*1*1*111	1	1	1	1	1	1	1	1	1	1	1	-15	1	1	1	-3	1	1	1	-3	1	1	1	-15	1	1	1	-3	1	1
111*1*111	1	1	1	1	1	1	1	1	1	1	1	-11	1	1	31	33	1	-17	-37	-19	1	1	1	-11	1	1	55	57	1	-89
1**11*111	1	1	1	1	1	1	1	1	10	1	1	-21	14	15	31	-27	-16	-32	39	17	43	-65	-45	-85	126	66	91	-241	-86	-95
11*11*111	1	1	1	1	1	1	1	1	1	1	1	-17	-25	-13	16	33	18	-23	-37	-19	22	45	24	-41	-99	-103	1	155	204	40
1*111*111	1	1	1	1	1	1	1	1	1	1	12	-9	-25	-20	16	13	1	-5	-18	-43	-27	12	47	39	-24	-129	-107	32	117	70
11111*111	1	1	1	1	1	1	1	1	-8	1	12	7	1	-20	-29	-23	1	22	39	21	-27	-76	-91	-49	51	131	127	8	-173	-305
1****1111	1	1	1	1	1	1	1	1	1	-4	1	13	1	-13	-4	1	18	1	-18	-4	1	23	1	-35	-4	27	28	-13	-57	-4
11***1111	1	1	1	1	1	1	1	1	1	6	1	1	-12	1	11	1	1	-17	1	6	1	1	1	1	-4	-12	1	29	1	-9
1*1**1111	1	1	1	1	1	1	1	1	10	6	-10	-23	14	15	11	-23	-50	-8	20	66	22	-54	-91	-15	171	144	-17	-265	-231	141
111**1111	1	1	1	1	1	1	1	1	1	6	1	1	1	-13	11	1	1	1	-18	6	1	1	1	1	-4	1	1	-13	30	-9
1**1*1111	1	1	1	1	1	1	1	1	10	-4	-10	-17	14	29	11	-23	-50	-14	58	76	1	-120	-114	39	221	170	-152	-391	-202	365
11*1*1111	1	1	1	1	1	1	1	1	1	6	12	-5	-12	-13	11	33	18	-23	-37	-14	22	34	24	-5	-29	-64	-53	43	146	105

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1*11*1111	1	1	1	1	1	1	1	1	1	6	1	-17	-12	-6	11	17	1	-23	-37	-14	15	45	24	-41	-79	-64	28	106	88	-45
1111*1111	1	1	1	1	1	1	1	1	-8	-4	1	-5	-12	-20	-34	-23	-16	-14	-18	-44	-69	-87	-91	-85	-79	-90	-143	-216	-289	-340
1***11111	1	1	1	1	1	1	1	1	1	1	-10	-5	14	1	1	1	-16	-5	20	1	1	-10	-22	19	26	-12	1	-27	-28	55
11**11111	1	1	1	1	1	1	1	1	1	11	1	-5	-12	1	1	17	1	-5	-18	-9	1	23	24	-5	-24	-38	1	29	59	5
1*1*11111	1	1	1	1	1	1	1	1	1	1	12	-5	-12	-6	1	1	18	13	-18	-19	-6	-10	24	43	1	-38	-26	-34	1	85
111*11111	1	1	1	1	1	1	1	1	-8	1	12	7	1	-20	-29	-23	1	22	39	21	-27	-76	-91	-49	51	131	127	8	-173	-305
1**111111	1	1	1	1	1	1	1	1	1	11	1	-11	1	-6	1	1	18	1	-18	-9	-6	23	1	13	1	-25	-26	-6	59	11
11*111111	1	1	1	1	1	1	1	1	-8	1	12	13	1	-6	-14	-23	-16	10	39	41	15	-32	-68	-67	-24	53	127	134	30	-134
1*1111111	1	1	1	1	1	1	1	1	-8	1	12	1	1	1	1	-7	-16	-8	20	21	1	-10	1	-7	-24	-25	19	57	30	-29
111111111	1	1	1	1	1	1	1	1	1	-9	1	1	1	1	1	1	1	1	1	-9	1	1	1	1	1	1	1	1	1	-9
1*******1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-512	1	1	1
11******1	1	1	1	1	1	1	1	1	1	1	-10	4	14	1	-17	-47	1	76	-18	1	40	-32	1	76	-24	-90	106	57	59	-11
1*1******1	1	1	1	1	1	1	1	1	1	1	1	0	14	29	-2	-3	18	76	-18	37	-2	1	1	48	-24	-38	106	25	-28	-26
111******1	1	1	1	1	1	1	1	1	1	-4	12	-3	-12	1	11	-3	1	1	20	-8	-20	-10	1	21	-29	-38	28	25	-28	-79
1**1*****1	1	1	1	1	1	1	1	1	1	1	-10	-27	14	-6	1	13	18	64	-18	37	-216	-54	1	-27	-24	-90	343	102	-115	1
11*1*****1	1	1	1	1	1	1	1	1	1	-4	1	-12	-38	1	-7	-19	-16	13	20	-28	-2	45	70	-12	-54	-38	61	-59	-144	-61
1*11*****1	1	1	1	1	1	1	1	1	1	-4	12	-8	-12	8	-7	1	-33	31	20	-44	-9	-10	47	-32	-54	40	7	-20	-86	-31
1111*****1	1	1	1	1	1	1	1	1	1	1	-10	1	1	-13	1	17	1	1	-18	1	22	-10	1	1	-24	1	28	-13	1	1
1***1****1	1	1	1	1	1	1	1	1	1	11	-10	4	1	1	-17	-7	18	76	-18	11	-2	-54	1	-4	26	1	106	1	30	-151
11**1****1	1	1	1	1	1	1	1	1	1	1	12	-2	-12	-6	28	-39	-16	7	20	21	-100	-32	93	-34	-49	-38	88	190	-202	-122
1*1*1****1	1	1	1	1	1	1	1	1	1	1	1	9	-12	1	16	29	-16	1	1	17	-20	1	1	9	1	-12	1	-3	59	-44
111*1****1	1	1	1	1	1	1	1	1	1	-4	1	15	14	-20	-19	29	35	-14	-56	-28	92	89	-68	-153	-4	196	154	-248	-347	125
1**11****1	1	1	1	1	1	1	1	1	1	1	12	-12	-12	-6	-2	-19	-50	13	58	17	5	-10	70	36	-74	-38	7	18	-86	-206
11*11****1	1	1	1	1	1	1	1	1	1	-14	-10	18	14	-13	-37	13	18	25	-56	-58	40	78	24	-102	-129	66	205	11	-260	-287
1*111****1	1	1	1	1	1	1	1	1	1	-14	1	25	14	-20	-19	33	52	-8	-75	-34	92	111	-68	-191	-29	274	208	-300	-492	121
11111****1	1	1	1	1	1	1	1	1	1	1	1	-5	14	1	-14	1	1	-5	20	1	-20	1	1	-5	26	-12	-26	29	1	-20
1****1***1	1	1	1	1	1	1	1	1	1	11	-10	4	1	1	-17	-7	18	76	-18	11	-2	-54	1	-4	26	1	106	1	30	-151
11***1***1	1	1	1	1	1	1	1	1	1	1	12	1	1	-20	16	-23	1	1	1	1	-13	-32	24	17	-24	1	28	-20	30	-74
1*1**1***1	1	1	1	1	1	1	1	1	1	1	12	-6	-12	1	28	13	-33	7	58	-23	-86	12	70	-6	-149	-38	115	53	-202	-122
111**1***1	1	1	1	1	1	1	1	1	1	-14	1	-9	14	-6	-19	13	18	-14	1	-38	-13	45	1	-57	21	14	19	46	-86	-95
1**1*1***1	1	1	1	1	1	1	1	1	1	1	1	-12	-12	1	-2	-19	-50	31	39	-3	-2	45	47	-36	-49	66	34	-87	-115	4
11*1*1***1	1	1	1	1	1	1	1	1	1	-4	-10	-3	1	-13	-49	13	35	-8	-56	-48	1	-10	-68	-99	-29	-25	-8	-73	-202	-259
1*11*1***1	1	1	1	1	1	1	1	1	1	-14	-10	10	27	-27	-37	-7	18	43	-37	-74	-2	78	47	-54	-179	-51	286	141	-260	-377
1111*1***1	1	1	1	1	1	1	1	1	1	1	-10	-17	1	29	1	-23	-16	-23	39	61	1	-76	-45	-9	101	131	-80	-223	-86	145
1***11***1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-16	1	-18	1	1	1	1	1	1	1	1	1	1	1
11**11***1	1	1	1	1	1	1	1	1	1	1	-10	-17	1	15	-14	1	1	-5	1	1	1	-10	-22	-17	26	1	-26	15	59	-20
1*1*11***1	1	1	1	1	1	1	1	1	1	-9	-10	3	1	-13	-44	-11	18	-14	-18	-53	-20	-10	24	-85	-99	-51	19	-17	-144	-210
111*11***1	1	1	1	1	1	1	1	1	1	6	-10	-3	1	29	-4	-11	-33	1	39	22	-20	-76	1	29	96	-25	-53	-115	30	126
1**111***1	1	1	1	1	1	1	1	1	1	1	-10	21	14	1	-14	21	1	-8	-18	-43	1	12	24	-43	-49	-64	73	25	-86	-74
11*111***1	1	1	1	1	1	1	1	1	1	-4	-21	-21	1	8	-19	-43	-33	-23	1	-8	-69	-87	-91	-37	-4	-77	-242	-276	-115	-55
1*1111***1	1	1	1	1	1	1	1	1	1	6	-10	-17	1	1	11	1	-33	-5	1	26	22	-54	-45	7	21	53	1	-111	-57	15

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
111111***1	1	1	1	1	1	1	1	1	1	1	1	-11	1	8	1	1	1	1	-18	1	15	1	1	-11	1	1	1	8	1	1
1*****1**1	1	1	1	1	1	1	1	1	1	1	-10	-27	14	-6	1	13	18	64	-18	37	-216	-54	1	-27	-24	-90	343	102	-115	1
11****1**1	1	1	1	1	1	1	1	1	1	1	12	-12	-12	1	-2	-3	-16	31	20	-23	-2	12	93	-12	-49	40	7	-31	-115	-26
1*1***1**1	1	1	1	1	1	1	1	1	1	-4	12	-12	1	8	-7	-3	-50	13	58	-28	-9	-10	47	-36	-54	53	7	-24	-57	-31
111***1**1	1	1	1	1	1	1	1	1	1	-14	-10	21	27	-27	-49	13	52	-8	-113	-18	85	144	-68	-195	-179	183	316	-87	-579	-209
1**1**1**1	1	1	1	1	1	1	1	1	1	-4	1	1	1	1	-64	1	1	1	1	-4	1	1	1	1	-4	1	1	1	1	-64
11*1**1**1	1	1	1	1	1	1	1	1	1	-14	1	4	1	15	-7	1	1	4	-18	-34	-2	1	1	28	21	1	-2	-13	-57	-71
1*11**1**1	1	1	1	1	1	1	1	1	1	1	1	4	14	1	-2	-7	18	4	1	1	-23	23	-22	12	-24	-12	-2	-27	30	-86
1111**1**1	1	1	1	1	1	1	1	1	1	1	-10	1	1	1	1	-7	1	-17	20	1	1	-10	1	9	1	27	-26	1	-28	31
1***1*1**1	1	1	1	1	1	1	1	1	1	1	1	-12	-12	1	-2	-19	-50	31	39	-3	-2	45	47	-36	-49	66	34	-87	-115	4
11**1*1**1	1	1	1	1	1	1	1	1	1	-9	-10	18	1	1	-47	13	35	-11	-37	-13	19	34	1	-78	-49	-25	151	-31	-173	-147
1*1*1*1**1	1	1	1	1	1	1	1	1	1	-4	1	-3	14	15	-19	-3	18	1	-18	-8	22	1	-22	-3	21	-12	-53	11	59	-49
111*1*1**1	1	1	1	1	1	1	1	1	1	6	-10	-21	1	29	11	-35	-16	13	39	2	-20	-10	1	-21	46	79	-26	-143	-57	195
1**11*1**1	1	1	1	1	1	1	1	1	1	-4	-10	4	14	1	-7	1	18	4	-18	-4	-23	-10	1	4	-4	14	-2	1	1	-91
11*11*1**1	1	1	1	1	1	1	1	1	1	6	1	-2	1	8	23	1	-16	7	1	6	-9	1	-22	-50	-4	1	7	-48	-57	-27
1*111*1**1	1	1	1	1	1	1	1	1	1	1	-10	1	14	43	1	1	-16	-17	1	1	1	-10	1	25	51	-12	-80	-153	-57	31
11111*1**1	1	1	1	1	1	1	1	1	1	11	1	-5	-25	-6	16	33	18	-14	-56	-49	15	89	93	-5	-124	-181	-44	190	320	140
1****11**1	1	1	1	1	1	1	1	1	1	1	12	-12	-12	-6	-2	-19	-50	13	58	17	5	-10	70	36	-74	-38	7	18	-86	-206
11***11**1	1	1	1	1	1	1	1	1	1	1	1	-3	14	1	-14	-19	35	19	-37	-23	1	1	1	-3	1	-12	1	25	30	-74
1*1**11**1	1	1	1	1	1	1	1	1	1	-4	1	18	1	-20	-37	21	18	-11	-37	-28	47	45	-22	-46	-4	53	97	-52	-144	-37
111**11**1	1	1	1	1	1	1	1	1	1	-4	-10	3	-12	29	-19	5	-50	31	1	32	-41	-10	-22	11	71	-38	28	-171	146	-115
1**1*11**1	1	1	1	1	1	1	1	1	1	-4	-10	4	14	1	-7	1	18	4	-18	-4	-23	-10	1	4	-4	14	-2	1	1	-91
11*1*11**1	1	1	1	1	1	1	1	1	1	6	1	1	1	8	-4	-63	-16	19	20	26	-27	-21	-45	1	46	79	55	-48	-28	36
1*11*11**1	1	1	1	1	1	1	1	1	1	1	-10	-2	-12	15	13	1	1	7	1	21	-2	-10	-22	-50	1	14	7	15	-28	-17
1111*11**1	1	1	1	1	1	1	1	1	1	11	12	-5	-38	-6	16	49	35	-14	-75	-89	-6	122	162	67	-174	-350	-179	218	610	470
1***111**1	1	1	1	1	1	1	1	1	1	1	-10	21	14	1	-14	21	1	-8	-18	-43	1	12	24	-43	-49	-64	73	25	-86	-74
11**111**1	1	1	1	1	1	1	1	1	1	1	-10	3	-12	8	1	-11	1	-23	20	-3	-6	-10	24	-13	-24	14	28	-24	-57	85
1*1*111**1	1	1	1	1	1	1	1	1	1	6	1	-21	1	15	11	-19	-33	-23	20	2	1	-43	-45	-45	21	27	28	-101	-115	-45
111*111**1	1	1	1	1	1	1	1	1	1	6	12	-15	-12	-6	26	13	1	1	-18	2	-6	34	47	9	-29	-90	28	74	88	6
1**1111**1	1	1	1	1	1	1	1	1	1	-4	-10	1	1	-6	-19	1	1	1	1	-4	-6	-10	1	1	-4	1	1	-6	1	-19
11*1111**1	1	1	1	1	1	1	1	1	1	6	1	-5	-12	29	26	1	-16	-14	1	26	22	1	-45	-53	-29	40	91	-27	-144	-120
1*11111**1	1	1	1	1	1	1	1	1	1	11	12	-5	-12	-6	1	9	35	4	-18	-49	-27	12	70	35	1	-90	-98	-62	117	185
1111111**1	1	1	1	1	1	1	1	1	1	1	12	1	-12	1	1	-7	1	1	20	1	-20	-10	1	17	1	-12	28	1	-28	-29
1******1*1	1	1	1	1	1	1	1	1	1	1	1	0	14	29	-2	-3	18	76	-18	37	-2	1	1	48	-24	-38	106	25	-28	-26
11*****1*1	1	1	1	1	1	1	1	1	1	-4	12	-12	-12	1	-7	-3	-16	13	58	-28	-86	-54	139	84	-129	-90	88	81	-86	-91
1*1****1*1	1	1	1	1	1	1	1	1	1	1	1	1	-12	1	1	1	1	1	-18	1	1	1	-45	1	1	-12	1	1	1	1
111****1*1	1	1	1	1	1	1	1	1	1	-4	1	13	14	-13	-19	17	35	-8	-18	-24	1	67	1	-59	-29	66	73	-41	-144	-49
1**1***1*1	1	1	1	1	1	1	1	1	1	-4	12	-12	1	8	-7	-3	-50	13	58	-28	-9	-10	47	-36	-54	53	7	-24	-57	-31
11*1***1*1	1	1	1	1	1	1	1	1	1	-9	-10	-12	1	1	-2	-19	1	67	-56	-133	-2	78	1	-108	-24	105	43	-143	-115	-6
1*11***1*1	1	1	1	1	1	1	1	1	1	-14	-10	1	1	8	-34	1	18	10	-56	-74	-6	-10	1	-47	-4	27	73	-76	-144	-164
1111***1*1	1	1	1	1	1	1	1	1	1	1	-10	-23	-12	15	16	17	-33	-17	20	21	22	-10	-45	-71	-24	66	109	15	-144	-134

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1***1**1*1	1	1	1	1	1	1	1	1	1	1	12	-6	-12	1	28	13	-33	7	58	-23	-86	12	70	-6	-149	-38	115	53	-202	-122
11**1**1*1	1	1	1	1	1	1	1	1	1	-14	1	18	14	-6	-37	-3	69	-38	-113	-18	47	45	-22	-174	-54	118	187	-66	-318	-197
1*1*1**1*1	1	1	1	1	1	1	1	1	1	1	1	-5	14	15	-14	-23	1	-5	20	1	-20	1	24	3	1	-12	-53	15	88	-20
111*1**1*1	1	1	1	1	1	1	1	1	1	-4	-10	-5	14	22	-19	-23	-16	13	39	16	-55	-54	47	75	46	-38	-161	-34	204	125
1**11**1*1	1	1	1	1	1	1	1	1	1	-4	1	18	1	-20	-37	21	18	-11	-37	-28	47	45	-22	-46	-4	53	97	-52	-144	-37
11*11**1*1	1	1	1	1	1	1	1	1	1	1	-10	-30	1	43	-2	-59	-33	7	58	17	-23	-54	-68	-22	176	183	-182	-437	-57	478
1*111**1*1	1	1	1	1	1	1	1	1	1	6	1	-5	1	-6	11	1	-50	-41	1	46	71	-65	-160	-5	146	183	28	-286	-173	195
11111**1*1	1	1	1	1	1	1	1	1	1	11	12	-17	-25	1	16	17	18	-32	-56	-29	22	78	93	-17	-149	-129	37	197	262	50
1****1*1*1	1	1	1	1	1	1	1	1	1	1	1	9	-12	1	16	29	-16	1	1	17	-20	1	1	9	1	-12	1	-3	59	-44
11***1*1*1	1	1	1	1	1	1	1	1	1	-4	-10	-3	1	-20	-19	-3	18	-8	-75	-48	8	-10	-45	-75	-54	-25	-35	-52	-86	-199
1*1**1*1*1	1	1	1	1	1	1	1	1	1	1	1	-5	14	15	-14	-23	1	-5	20	1	-20	1	24	3	1	-12	-53	15	88	-20
111**1*1*1	1	1	1	1	1	1	1	1	1	6	1	-5	14	22	-4	-23	-33	13	39	6	-34	-43	1	75	71	-12	-80	-90	30	210
1**1*1*1*1	1	1	1	1	1	1	1	1	1	-4	1	-3	14	15	-19	-3	18	1	-18	-8	22	1	-22	-3	21	-12	-53	11	59	-49
11*1*1*1*1	1	1	1	1	1	1	1	1	1	1	1	-15	-12	1	16	-3	-16	1	20	-3	-20	1	24	-15	-24	-12	28	25	-28	-44
1*11*1*1*1	1	1	1	1	1	1	1	1	1	-4	1	-5	-12	1	-4	1	1	-5	1	-4	1	1	1	-5	-4	-12	1	1	1	-10
1111*1*1*1	1	1	1	1	1	1	1	1	1	1	1	-5	-12	1	1	1	18	-5	-18	1	1	1	24	-5	-24	-12	1	1	30	25
1***11*1*1	1	1	1	1	1	1	1	1	1	-9	-10	3	1	-13	-44	-11	18	-14	-18	-53	-20	-10	24	-85	-99	-51	19	-17	-144	-210
11**11*1*1	1	1	1	1	1	1	1	1	1	6	-10	-21	14	29	11	-59	-33	31	58	-18	-62	-54	-22	35	96	40	-188	-339	59	495
1*1*11*1*1	1	1	1	1	1	1	1	1	1	1	1	1	-12	1	1	1	1	1	-18	1	1	1	1	1	1	-12	1	1	1	1
111*11*1*1	1	1	1	1	1	1	1	1	1	6	1	-11	-25	15	26	17	-16	-44	-18	26	43	45	-22	-83	-79	1	172	155	-57	-264
1**111*1*1	1	1	1	1	1	1	1	1	1	6	1	-21	1	15	11	-19	-33	-23	20	2	1	-43	-45	-45	21	27	28	-101	-115	-45
11*111*1*1	1	1	1	1	1	1	1	1	1	1	1	-9	-12	-20	16	13	18	-32	-18	-23	15	1	70	-9	-49	-116	37	32	117	10
1*1111*1*1	1	1	1	1	1	1	1	1	1	6	1	1	-25	1	11	-7	1	1	1	6	-20	1	24	-7	-4	-25	1	29	-28	-9
111111*1*1	1	1	1	1	1	1	1	1	1	1	1	13	1	-6	-14	-7	1	1	20	21	-6	-21	-22	-19	1	27	55	22	-28	-44
1*****11*1	1	1	1	1	1	1	1	1	1	-4	12	-8	-12	8	-7	1	-33	31	20	-44	-9	-10	47	-32	-54	40	7	-20	-86	-31
11****11*1	1	1	1	1	1	1	1	1	1	-14	1	4	14	1	-7	17	18	13	-56	-74	19	45	1	-20	-4	66	43	-27	-144	-131
1*1***11*1	1	1	1	1	1	1	1	1	1	-14	-10	1	1	8	-34	1	18	10	-56	-74	-6	-10	1	-47	-4	27	73	-76	-144	-164
111***11*1	1	1	1	1	1	1	1	1	1	6	-10	-11	-12	15	26	-15	-50	19	58	-14	-20	-10	1	13	-29	-12	55	-41	-115	36
1**1**11*1	1	1	1	1	1	1	1	1	1	1	1	4	14	1	-2	-7	18	4	1	1	-23	23	-22	12	-24	-12	-2	-27	30	-86
11*1**11*1	1	1	1	1	1	1	1	1	1	1	1	4	1	1	13	-23	-50	-41	20	21	-2	1	-22	36	51	79	88	-27	-86	-41
1*11**11*1	1	1	1	1	1	1	1	1	1	1	-10	1	1	29	-44	1	-50	37	-18	41	-41	-10	-22	1	51	-51	1	-139	175	-104
1111**11*1	1	1	1	1	1	1	1	1	1	11	12	1	-25	-13	16	33	1	-44	-56	-29	-20	34	70	49	-49	-129	-44	183	320	176
1***1*11*1	1	1	1	1	1	1	1	1	1	-14	-10	10	27	-27	-37	-7	18	43	-37	-74	-2	78	47	-54	-179	-51	286	141	-260	-377
11**1*11*1	1	1	1	1	1	1	1	1	1	6	1	-26	1	29	-7	-55	-50	25	39	-14	-23	-43	1	6	96	131	-128	-251	-86	393
1*1*1*11*1	1	1	1	1	1	1	1	1	1	-4	1	-5	-12	1	-4	1	1	-5	1	-4	1	1	1	-5	-4	-12	1	1	1	-10
111*1*11*1	1	1	1	1	1	1	1	1	1	6	1	-5	-38	1	41	33	-16	-68	-37	46	64	45	-22	-77	-79	-64	118	281	117	-315
1**11*11*1	1	1	1	1	1	1	1	1	1	1	-10	-2	-12	15	13	1	1	7	1	21	-2	-10	-22	-50	1	14	7	15	-28	-17
11*11*11*1	1	1	1	1	1	1	1	1	1	1	1	10	-25	-20	13	1	1	-2	1	-19	-9	1	1	10	26	-25	-83	8	30	13
1*111*11*1	1	1	1	1	1	1	1	1	1	1	1	-5	-12	-13	16	-7	18	-14	1	-19	22	-21	47	-37	26	-64	37	-41	88	-80
11111*11*1	1	1	1	1	1	1	1	1	1	1	12	7	-12	-20	-14	-7	18	31	20	-19	-48	-54	-22	47	101	66	-26	-132	-173	-80
1****111*1	1	1	1	1	1	1	1	1	1	-14	1	25	14	-20	-19	33	52	-8	-75	-34	92	111	-68	-191	-29	274	208	-300	-492	121

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
11***111*1	1	1	1	1	1	1	1	1	1	-4	-10	-11	14	15	11	-47	-33	19	39	16	-41	-76	1	61	96	-12	-161	-153	88	281
1*1**111*1	1	1	1	1	1	1	1	1	1	6	1	-5	1	-6	11	1	-50	-41	1	46	71	-65	-160	-5	146	183	28	-286	-173	195
111**111*1	1	1	1	1	1	1	1	1	1	6	1	-5	-12	-13	26	33	-33	-59	-37	26	64	1	-68	-53	-29	14	55	71	59	-120
1**1*111*1	1	1	1	1	1	1	1	1	1	1	-10	1	14	43	1	1	-16	-17	1	1	1	-10	1	25	51	-12	-80	-153	-57	31
11*1*111*1	1	1	1	1	1	1	1	1	1	1	1	1	-25	-6	16	33	-16	-26	1	1	15	23	24	-23	-74	-25	37	78	88	-74
1*11*111*1	1	1	1	1	1	1	1	1	1	1	1	-5	-12	-13	16	-7	18	-14	1	-19	22	-21	47	-37	26	-64	37	-41	88	-80
1111*111*1	1	1	1	1	1	1	1	1	1	1	12	19	-12	-34	-29	-7	35	49	39	-19	-90	-98	-45	59	201	196	1	-286	-434	-245
1***1111*1	1	1	1	1	1	1	1	1	1	6	-10	-17	1	1	11	1	-33	-5	1	26	22	-54	-45	7	21	53	1	-111	-57	15
11**1111*1	1	1	1	1	1	1	1	1	1	6	12	-17	-25	-6	26	33	1	-32	-75	-34	36	78	93	-65	-179	-181	37	330	291	0
1*1*1111*1	1	1	1	1	1	1	1	1	1	6	1	1	-25	1	11	-7	1	1	1	6	-20	1	24	-7	-4	-25	1	29	-28	-9
111*1111*1	1	1	1	1	1	1	1	1	1	-4	12	13	-12	-20	-34	9	18	19	20	-24	-27	-54	-45	29	71	118	1	-160	-173	-124
1**11111*1	1	1	1	1	1	1	1	1	1	11	12	-5	-12	-6	1	9	35	4	-18	-49	-27	12	70	35	1	-90	-98	-62	117	185
11*11111*1	1	1	1	1	1	1	1	1	1	1	12	7	-12	-27	1	-7	18	13	20	-19	-20	-54	1	23	101	14	-26	-111	-57	-65
1*111111*1	1	1	1	1	1	1	1	1	1	1	1	1	-12	-6	1	1	1	1	1	1	-6	1	1	1	1	-12	1	-6	1	1
11111111*1	1	1	1	1	1	1	1	1	1	-9	1	13	1	1	1	1	1	-8	-18	-9	22	23	1	-11	1	1	-8	-27	-28	21
1*******11	1	1	1	1	1	1	1	1	1	1	-10	4	14	1	-17	-47	1	76	-18	1	40	-32	1	76	-24	-90	106	57	59	-11
11******11	1	1	1	1	1	1	1	1	1	-4	1	1	1	1	-4	1	1	1	1	-4	1	1	-45	1	-29	1	1	1	1	-4
1*1*****11	1	1	1	1	1	1	1	1	1	-4	12	-12	-12	1	-7	-3	-16	13	58	-28	-86	-54	139	84	-129	-90	88	81	-86	-91
111*****11	1	1	1	1	1	1	1	1	1	1	-10	9	14	15	1	-3	-16	19	1	-3	-20	12	24	9	-49	-38	-26	11	1	1
1**1****11	1	1	1	1	1	1	1	1	1	1	12	-12	-12	1	-2	-3	-16	31	20	-23	-2	12	93	-12	-49	40	7	-31	-115	-26
11*1****11	1	1	1	1	1	1	1	1	1	-4	-10	21	1	1	-19	-19	18	10	-37	-28	22	78	24	-147	-79	-25	46	53	-115	-79
1*11****11	1	1	1	1	1	1	1	1	1	-14	1	4	14	1	-7	17	18	13	-56	-74	19	45	1	-20	-4	66	43	-27	-144	-131
1111****11	1	1	1	1	1	1	1	1	1	1	-10	-11	-12	1	16	17	-16	-17	1	21	22	-10	-22	-35	-24	14	55	29	-28	-74
1***1***11	1	1	1	1	1	1	1	1	1	1	12	1	1	-20	16	-23	1	1	1	1	-13	-32	24	17	-24	1	28	-20	30	-74
11**1***11	1	1	1	1	1	1	1	1	1	-14	-10	-17	14	-20	-4	9	1	-14	1	-114	-13	-10	-22	-1	46	-64	73	-48	-144	-140
1*1*1***11	1	1	1	1	1	1	1	1	1	-4	-10	-3	1	-20	-19	-3	18	-8	-75	-48	8	-10	-45	-75	-54	-25	-35	-52	-86	-199
111*1***11	1	1	1	1	1	1	1	1	1	1	-10	-21	14	-6	1	-19	-16	-5	39	-23	-13	-54	-45	3	51	-64	1	-66	30	85
1**11***11	1	1	1	1	1	1	1	1	1	1	1	-3	14	1	-14	-19	35	19	-37	-23	1	1	1	-3	1	-12	1	25	30	-74
11*11***11	1	1	1	1	1	1	1	1	1	6	-10	3	1	15	11	13	-33	13	20	42	22	-32	-22	3	46	27	28	-101	-28	-45
1*111***11	1	1	1	1	1	1	1	1	1	-4	-10	-11	14	15	11	-47	-33	19	39	16	-41	-76	1	61	96	-12	-161	-153	88	281
11111***11	1	1	1	1	1	1	1	1	1	1	12	-5	1	-13	1	1	18	-5	1	-19	1	-10	24	-5	26	-25	1	-41	30	-5
1****1**11	1	1	1	1	1	1	1	1	1	1	12	-2	-12	-6	28	-39	-16	7	20	21	-100	-32	93	-34	-49	-38	88	190	-202	-122
11***1**11	1	1	1	1	1	1	1	1	1	-14	-10	-17	14	-20	-4	9	1	-14	1	-114	-13	-10	-22	-1	46	-64	73	-48	-144	-140
1*1**1**11	1	1	1	1	1	1	1	1	1	-14	1	18	14	-6	-37	-3	69	-38	-113	-18	47	45	-22	-174	-54	118	187	-66	-318	-197
111**1**11	1	1	1	1	1	1	1	1	1	1	1	3	14	22	-14	-3	-16	31	39	17	-76	23	1	3	1	-64	-80	74	30	-50
1**1*1**11	1	1	1	1	1	1	1	1	1	-9	-10	18	1	1	-47	13	35	-11	-37	-13	19	34	1	-78	-49	-25	151	-31	-173	-147
11*1*1**11	1	1	1	1	1	1	1	1	1	6	-10	-21	1	29	11	-35	-16	31	96	22	-62	-76	1	51	146	1	-134	-171	1	345
1*11*1**11	1	1	1	1	1	1	1	1	1	6	1	-26	1	29	-7	-55	-50	25	39	-14	-23	-43	1	6	96	131	-128	-251	-86	393
1111*1**11	1	1	1	1	1	1	1	1	1	11	12	-17	-25	15	31	25	1	-86	-94	11	85	78	24	-153	-249	1	334	407	204	-385
1***11**11	1	1	1	1	1	1	1	1	1	1	-10	-17	1	15	-14	1	1	-5	1	1	1	-10	-22	-17	26	1	-26	15	59	-20
11**11**11	1	1	1	1	1	1	1	1	1	-4	1	1	1	1	-4	1	-33	1	1	-4	1	1	1	1	-4	1	1	1	1	-4

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1*1*11**11	1	1	1	1	1	1	1	1	1	6	-10	-21	14	29	11	-59	-33	31	58	-18	-62	-54	-22	35	96	40	-188	-339	59	495
111*11**11	1	1	1	1	1	1	1	1	1	11	1	-3	1	1	16	5	-16	-26	-18	-13	1	1	-22	-43	-24	27	64	53	30	-4
1**111**11	1	1	1	1	1	1	1	1	1	1	-10	3	-12	8	1	-11	1	-23	20	-3	-6	-10	24	-13	-24	14	28	-24	-57	85
11*111**11	1	1	1	1	1	1	1	1	1	6	1	-3	1	-6	26	5	1	-17	1	2	-6	23	-22	5	-54	1	1	18	30	-84
1*1111**11	1	1	1	1	1	1	1	1	1	6	12	-17	-25	-6	26	33	1	-32	-75	-34	36	78	93	-65	-179	-181	37	330	291	0
111111**11	1	1	1	1	1	1	1	1	1	1	12	1	1	-20	1	1	1	19	1	1	-27	-10	1	1	51	1	1	-48	-28	1
1*****1*11	1	1	1	1	1	1	1	1	1	-4	1	-12	-38	1	-7	-19	-16	13	20	-28	-2	45	70	-12	-54	-38	61	-59	-144	-61
11****1*11	1	1	1	1	1	1	1	1	1	-4	-10	21	1	1	-19	-19	18	10	-37	-28	22	78	24	-147	-79	-25	46	53	-115	-79
1*1***1*11	1	1	1	1	1	1	1	1	1	-9	-10	-12	1	1	-2	-19	1	67	-56	-133	-2	78	1	-108	-24	105	43	-143	-115	-6
111***1*11	1	1	1	1	1	1	1	1	1	1	-21	-27	-12	1	16	13	-33	-17	39	17	-41	-153	-114	-51	1	14	28	-31	-86	16
1**1**1*11	1	1	1	1	1	1	1	1	1	-14	1	4	1	15	-7	1	1	4	-18	-34	-2	1	1	28	21	1	-2	-13	-57	-71
11*1**1*11	1	1	1	1	1	1	1	1	1	-4	1	1	-12	1	-19	1	1	37	39	-4	1	1	-22	-47	-29	-12	1	1	1	-19
1*11**1*11	1	1	1	1	1	1	1	1	1	1	1	4	1	1	13	-23	-50	-41	20	21	-2	1	-22	36	51	79	88	-27	-86	-41
1111**1*11	1	1	1	1	1	1	1	1	1	11	1	1	-12	1	1	9	-16	-62	-37	-29	22	23	24	-15	1	66	145	169	88	-19
1***1*1*11	1	1	1	1	1	1	1	1	1	-4	-10	-3	1	-13	-49	13	35	-8	-56	-48	1	-10	-68	-99	-29	-25	-8	-73	-202	-259
11**1*1*11	1	1	1	1	1	1	1	1	1	6	-10	-21	1	29	11	-35	-16	31	96	22	-62	-76	1	51	146	1	-134	-171	1	345
1*1*1*1*11	1	1	1	1	1	1	1	1	1	1	1	-15	-12	1	16	-3	-16	1	20	-3	-20	1	24	-15	-24	-12	28	25	-28	-44
111*1*1*11	1	1	1	1	1	1	1	1	1	1	1	-9	-12	-13	16	29	1	-95	-18	57	43	-43	-45	-57	26	-12	-26	11	117	-50
1**11*1*11	1	1	1	1	1	1	1	1	1	6	1	1	1	8	-4	-63	-16	19	20	26	-27	-21	-45	1	46	79	55	-48	-28	36
11*11*1*11	1	1	1	1	1	1	1	1	1	6	1	-5	-25	-6	41	49	1	-50	-37	6	57	67	1	-77	-104	-51	64	190	146	-165
1*111*1*11	1	1	1	1	1	1	1	1	1	1	1	1	-25	-6	16	33	-16	-26	1	1	15	23	24	-23	-74	-25	37	78	88	-74
11111*1*11	1	1	1	1	1	1	1	1	1	1	12	19	-12	-20	-14	1	18	49	20	-39	-48	-32	-22	43	126	66	-53	-132	-173	-110
1****11*11	1	1	1	1	1	1	1	1	1	-14	-10	18	14	-13	-37	13	18	25	-56	-58	40	78	24	-102	-129	66	205	11	-260	-287
11***11*11	1	1	1	1	1	1	1	1	1	6	-10	3	1	15	11	13	-33	13	20	42	22	-32	-22	3	46	27	28	-101	-28	-45
1*1**11*11	1	1	1	1	1	1	1	1	1	1	-10	-30	1	43	-2	-59	-33	7	58	17	-23	-54	-68	-22	176	183	-182	-437	-57	478
111**11*11	1	1	1	1	1	1	1	1	1	11	1	-21	-12	1	16	21	-16	-68	-18	27	43	23	-22	-109	-74	40	145	137	-57	-280
1**1*11*11	1	1	1	1	1	1	1	1	1	6	1	-2	1	8	23	1	-16	7	1	6	-9	1	-22	-50	-4	1	7	-48	-57	-27
11*1*11*11	1	1	1	1	1	1	1	1	1	6	1	-5	-25	-6	41	49	1	-50	-37	6	57	67	1	-77	-104	-51	64	190	146	-165
1*11*11*11	1	1	1	1	1	1	1	1	1	1	1	10	-25	-20	13	1	1	-2	1	-19	-9	1	1	10	26	-25	-83	8	30	13
1111*11*11	1	1	1	1	1	1	1	1	1	1	1	7	-12	-20	-14	1	18	31	1	-39	-48	-21	24	55	51	-12	-80	-104	-57	40
1***111*11	1	1	1	1	1	1	1	1	1	-4	-21	-21	1	8	-19	-43	-33	-23	1	-8	-69	-87	-91	-37	-4	-77	-242	-276	-115	-55
11**111*11	1	1	1	1	1	1	1	1	1	6	1	-3	1	-6	26	5	1	-17	1	2	-6	23	-22	5	-54	1	1	18	30	-84
1*1*111*11	1	1	1	1	1	1	1	1	1	1	1	-9	-12	-20	16	13	18	-32	-18	-23	15	1	70	-9	-49	-116	37	32	117	10
111*111*11	1	1	1	1	1	1	1	1	1	1	1	9	1	-34	-29	-3	18	1	1	-3	-27	-43	-22	9	26	53	28	-94	-173	-89
1**1111*11	1	1	1	1	1	1	1	1	1	6	1	-5	-12	29	26	1	-16	-14	1	26	22	1	-45	-53	-29	40	91	-27	-144	-120
11*1111*11	1	1	1	1	1	1	1	1	1	-4	1	1	-12	-27	-19	1	1	1	1	-4	1	1	1	1	-4	-12	-53	-83	-57	-19
1*11111*11	1	1	1	1	1	1	1	1	1	1	12	7	-12	-27	1	-7	18	13	20	-19	-20	-54	1	23	101	14	-26	-111	-57	-65
1111111*11	1	1	1	1	1	1	1	1	1	-9	1	13	14	1	1	-7	-16	-26	-18	11	43	45	24	-19	-49	-64	-62	-27	59	141
1******111	1	1	1	1	1	1	1	1	1	-4	12	-3	-12	1	11	-3	1	1	20	-8	-20	-10	1	21	-29	-38	28	25	-28	-79
11*****111	1	1	1	1	1	1	1	1	1	1	-10	9	14	15	1	-3	-16	19	1	-3	-20	12	24	9	-49	-38	-26	11	1	1
1*1****111	1	1	1	1	1	1	1	1	1	-4	1	13	14	-13	-19	17	35	-8	-18	-24	1	67	1	-59	-29	66	73	-41	-144	-49

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
111****111	1	1	1	1	1	1	1	1	1	1	-10	1	1	1	1	1	1	1	-18	1	1	-10	1	1	1	1	1	1	1	1
1**1***111	1	1	1	1	1	1	1	1	1	-14	-10	21	27	-27	-49	13	52	-8	-113	-18	85	144	-68	-195	-179	183	316	-87	-579	-209
11*1***111	1	1	1	1	1	1	1	1	1	1	-21	-27	-12	1	16	13	-33	-17	39	17	-41	-153	-114	-51	1	14	28	-31	-86	16
1*11***111	1	1	1	1	1	1	1	1	1	6	-10	-11	-12	15	26	-15	-50	19	58	-14	-20	-10	1	13	-29	-12	55	-41	-115	36
1111***111	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-14	17	1	1	1	-19	1	1	1	1	1	1	1	1	1	-14
1***1**111	1	1	1	1	1	1	1	1	1	-14	1	-9	14	-6	-19	13	18	-14	1	-38	-13	45	1	-57	21	14	19	46	-86	-95
11**1**111	1	1	1	1	1	1	1	1	1	1	1	3	14	22	-14	-3	-16	31	39	17	-76	23	1	3	1	-64	-80	74	30	-50
1*1*1**111	1	1	1	1	1	1	1	1	1	6	1	-5	14	22	-4	-23	-33	13	39	6	-34	-43	1	75	71	-12	-80	-90	30	210
111*1**111	1	1	1	1	1	1	1	1	1	11	12	-5	1	8	1	9	-16	-68	-94	-69	-55	-10	1	-45	1	131	226	372	407	245
1**11**111	1	1	1	1	1	1	1	1	1	-4	-10	3	-12	29	-19	5	-50	31	1	32	-41	-10	-22	11	71	-38	28	-171	146	-115
11*11**111	1	1	1	1	1	1	1	1	1	11	1	-21	-12	1	16	21	-16	-68	-18	27	43	23	-22	-109	-74	40	145	137	-57	-280
1*111**111	1	1	1	1	1	1	1	1	1	6	1	-5	-12	-13	26	33	-33	-59	-37	26	64	1	-68	-53	-29	14	55	71	59	-120
11111**111	1	1	1	1	1	1	1	1	1	1	12	-5	1	1	-14	1	18	-5	1	1	-20	-10	24	-5	1	27	-26	-27	30	-20
1****1*111	1	1	1	1	1	1	1	1	1	-4	1	15	14	-20	-19	29	35	-14	-56	-28	92	89	-68	-153	-4	196	154	-248	-347	125
11***1*111	1	1	1	1	1	1	1	1	1	1	-10	-21	14	-6	1	-19	-16	-5	39	-23	-13	-54	-45	3	51	-64	1	-66	30	85
1*1**1*111	1	1	1	1	1	1	1	1	1	-4	-10	-5	14	22	-19	-23	-16	13	39	16	-55	-54	47	75	46	-38	-161	-34	204	125
111**1*111	1	1	1	1	1	1	1	1	1	11	12	-5	1	8	1	9	-16	-68	-94	-69	-55	-10	1	-45	1	131	226	372	407	245
1**1*1*111	1	1	1	1	1	1	1	1	1	6	-10	-21	1	29	11	-35	-16	13	39	2	-20	-10	1	-21	46	79	-26	-143	-57	195
11*1*1*111	1	1	1	1	1	1	1	1	1	1	1	-9	-12	-13	16	29	1	-95	-18	57	43	-43	-45	-57	26	-12	-26	11	117	-50
1*11*1*111	1	1	1	1	1	1	1	1	1	6	1	-5	-38	1	41	33	-16	-68	-37	46	64	45	-22	-77	-79	-64	118	281	117	-315
1111*1*111	1	1	1	1	1	1	1	1	1	1	12	7	-12	1	-14	1	52	13	-18	-19	-41	-10	47	31	1	-12	-26	-111	-28	100
1***11*111	1	1	1	1	1	1	1	1	1	6	-10	-3	1	29	-4	-11	-33	1	39	22	-20	-76	1	29	96	-25	-53	-115	30	126
11**11*111	1	1	1	1	1	1	1	1	1	11	1	-3	1	1	16	5	-16	-26	-18	-13	1	1	-22	-43	-24	27	64	53	30	-4
1*1*11*111	1	1	1	1	1	1	1	1	1	6	1	-11	-25	15	26	17	-16	-44	-18	26	43	45	-22	-83	-79	1	172	155	-57	-264
111*11*111	1	1	1	1	1	1	1	1	1	1	1	1	-12	-27	-14	33	69	73	20	-39	-62	-43	-22	1	1	-12	-53	-83	1	226
1**111*111	1	1	1	1	1	1	1	1	1	6	12	-15	-12	-6	26	13	1	1	-18	2	-6	34	47	9	-29	-90	28	74	88	6
11*111*111	1	1	1	1	1	1	1	1	1	1	1	9	1	-34	-29	-3	18	1	1	-3	-27	-43	-22	9	26	53	28	-94	-173	-89
1*1111*111	1	1	1	1	1	1	1	1	1	-4	12	13	-12	-20	-34	9	18	19	20	-24	-27	-54	-45	29	71	118	1	-160	-173	-124
111111*111	1	1	1	1	1	1	1	1	1	-9	1	13	14	8	-14	-23	-33	-26	1	31	57	45	1	-67	-124	-116	-35	92	204	216
1*****1111	1	1	1	1	1	1	1	1	1	1	-10	1	1	-13	1	17	1	1	-18	1	22	-10	1	1	-24	1	28	-13	1	1
11****1111	1	1	1	1	1	1	1	1	1	1	-10	-11	-12	1	16	17	-16	-17	1	21	22	-10	-22	-35	-24	14	55	29	-28	-74
1*1***1111	1	1	1	1	1	1	1	1	1	1	-10	-23	-12	15	16	17	-33	-17	20	21	22	-10	-45	-71	-24	66	109	15	-144	-134
111***1111	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-14	17	1	1	1	-19	1	1	1	1	1	1	1	1	1	-14
1**1**1111	1	1	1	1	1	1	1	1	1	1	-10	1	1	1	1	-7	1	-17	20	1	1	-10	1	9	1	27	-26	1	-28	31
11*1**1111	1	1	1	1	1	1	1	1	1	11	1	1	-12	1	1	9	-16	-62	-37	-29	22	23	24	-15	1	66	145	169	88	-19
1*11**1111	1	1	1	1	1	1	1	1	1	11	12	1	-25	-13	16	33	1	-44	-56	-29	-20	34	70	49	-49	-129	-44	183	320	176
1111**1111	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-20	1	1	1	1	1	1	1	1	1
1***1*1111	1	1	1	1	1	1	1	1	1	1	-10	-17	1	29	1	-23	-16	-23	39	61	1	-76	-45	-9	101	131	-80	-223	-86	145
11**1*1111	1	1	1	1	1	1	1	1	1	11	12	-17	-25	15	31	25	1	-86	-94	11	85	78	24	-153	-249	1	334	407	204	-385
1*1*1*1111	1	1	1	1	1	1	1	1	1	1	1	-5	-12	1	1	1	18	-5	-18	1	1	1	24	-5	-24	-12	1	1	30	25
111*1*1111	1	1	1	1	1	1	1	1	1	1	12	7	-12	1	-14	1	52	13	-18	-19	-41	-10	47	31	1	-12	-26	-111	-28	100

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1**11*1111	1	1	1	1	1	1	1	1	1	11	12	-5	-38	-6	16	49	35	-14	-75	-89	-6	122	162	67	-174	-350	-179	218	610	470
11*11*1111	1	1	1	1	1	1	1	1	1	1	1	7	-12	-20	-14	1	18	31	1	-39	-48	-21	24	55	51	-12	-80	-104	-57	40
1*111*1111	1	1	1	1	1	1	1	1	1	1	12	19	-12	-34	-29	-7	35	49	39	-19	-90	-98	-45	59	201	196	1	-286	-434	-245
11111*1111	1	1	1	1	1	1	1	1	1	-9	1	7	1	-6	-14	-39	-33	-14	1	11	15	-21	-68	-97	-99	-77	-8	50	30	-60
1****11111	1	1	1	1	1	1	1	1	1	1	1	-5	14	1	-14	1	1	-5	20	1	-20	1	1	-5	26	-12	-26	29	1	-20
11***11111	1	1	1	1	1	1	1	1	1	1	12	-5	1	-13	1	1	18	-5	1	-19	1	-10	24	-5	26	-25	1	-41	30	-5
1*1**11111	1	1	1	1	1	1	1	1	1	11	12	-17	-25	1	16	17	18	-32	-56	-29	22	78	93	-17	-149	-129	37	197	262	50
111**11111	1	1	1	1	1	1	1	1	1	1	12	-5	1	1	-14	1	18	-5	1	1	-20	-10	24	-5	1	27	-26	-27	30	-20
1**1*11111	1	1	1	1	1	1	1	1	1	11	1	-5	-25	-6	16	33	18	-14	-56	-49	15	89	93	-5	-124	-181	-44	190	320	140
11*1*11111	1	1	1	1	1	1	1	1	1	1	12	19	-12	-20	-14	1	18	49	20	-39	-48	-32	-22	43	126	66	-53	-132	-173	-110
1*11*11111	1	1	1	1	1	1	1	1	1	1	12	7	-12	-20	-14	-7	18	31	20	-19	-48	-54	-22	47	101	66	-26	-132	-173	-80
1111*11111	1	1	1	1	1	1	1	1	1	-9	1	7	1	-6	-14	-39	-33	-14	1	11	15	-21	-68	-97	-99	-77	-8	50	30	-60
1***111111	1	1	1	1	1	1	1	1	1	1	1	-11	1	8	1	1	1	1	-18	1	15	1	1	-11	1	1	1	8	1	1
11**111111	1	1	1	1	1	1	1	1	1	1	12	1	1	-20	1	1	1	19	1	1	-27	-10	1	1	51	1	1	-48	-28	1
1*1*111111	1	1	1	1	1	1	1	1	1	1	1	13	1	-6	-14	-7	1	1	20	21	-6	-21	-22	-19	1	27	55	22	-28	-44
111*111111	1	1	1	1	1	1	1	1	1	-9	1	13	14	8	-14	-23	-33	-26	1	31	57	45	1	-67	-124	-116	-35	92	204	216
1**1111111	1	1	1	1	1	1	1	1	1	1	12	1	-12	1	1	-7	1	1	20	1	-20	-10	1	17	1	-12	28	1	-28	-29
11*1111111	1	1	1	1	1	1	1	1	1	-9	1	13	14	1	1	-7	-16	-26	-18	11	43	45	24	-19	-49	-64	-62	-27	59	141
1*11111111	1	1	1	1	1	1	1	1	1	-9	1	13	1	1	1	1	1	-8	-18	-9	22	23	1	-11	1	1	-8	-27	-28	21
1111111111	1	1	1	1	1	1	1	1	1	1	-10	1	1	1	1	1	1	1	1	1	1	-10	1	1	1	1	1	1	1	1

Coefficients of Non-Saturated Monotone Minterm-Cyclic Functions

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Characterization
11*1	1	1	1	-3	1	4	-6	-3	7	1	-10	0	14	-6	-17	13	18	-23	-18	37	12	-54	1	72	-24	-90	61	102	-115	-101	
1*11	1	1	1	-3	1	4	-6	-3	7	1	-10	0	14	-6	-17	13	18	-23	-18	37	12	-54	1	72	-24	-90	61	102	-115	-101	Reverse of 11*1
1*1**1	1	1	1	1	1	4	1	-3	7	11	-10	0	14	1	-17	-3	18	-5	-37	7	40	-32	-45	48	51	-64	-47	109	59	-151	
1**1*1	1	1	1	1	1	4	1	-3	7	11	-10	0	14	1	-17	-3	18	-5	-37	7	40	-32	-45	48	51	-64	-47	109	59	-151	Reverse of 1*1**1
1*1***1	1	1	1	1	1	1	-6	-9	7	-1	-10	-16	14	-36	-17	-9	18	-49	-18	-1	12	-100	1	0	-24	-196	61	-36	-115	-289	Expanded 11*1
1***1*1	1	1	1	1	1	1	-6	-9	7	-1	-10	-16	14	-36	-17	-9	18	-49	-18	-1	12	-100	1	0	-24	-196	61	-36	-115	-289	Reverse of 1*1***1
1*1******1	1	1	1	1	1	1	1	1	1	1	1	0	14	29	-2	-3	18	76	-18	37	-2	1	1	48	-24	-38	106	25	-28	-26	Rotation of 1**1*1
1******1*1	1	1	1	1	1	1	1	1	1	1	1	0	14	29	-2	-3	18	76	-18	37	-2	1	1	48	-24	-38	106	25	-28	-26	Reverse of 1*1******1
11**1111*1	1	1	1	1	1	1	1	1	1	6	12	-17	-25	-6	26	33	1	-32	-75	-34	36	78	93	-65	-179	-181	37	330	291	0	
1*1111**11	1	1	1	1	1	1	1	1	1	6	12	-17	-25	-6	26	33	1	-32	-75	-34	36	78	93	-65	-179	-181	37	330	291	0	Reverse of 11**1111*1

Appendix C

Degree-n Coefficients of $\textsc{Match}_\alpha$ for All α

The following table contains the data generated by `mintermsFromNmFile` from Appendix A.1. Each row represents a single minterm-cyclic function $\textsc{Match}_\alpha$, given by the minterm $\alpha \in \{0, 1, *\}^k$ in the leftmost column. Note that this Appendix also includes non-monotone α. Each column represents an input size n. In each cell is the coefficient of the degree-n term in the unique multilinear polynomial representing $\textsc{Match}_\alpha$. This value is 0 if and only if the polynomial complexity of the function $\textsc{Match}_\alpha$ at input size n is strictly less than n. Thus, if a row (extended infinitely to the right) consists of all non-zero coefficients, then $\textsc{Match}_\alpha$ is *saturated*.

Self-Avoiding Minterms that Don't Contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1	-1	1
0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
01	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0
011	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
0111	0	0	0	-4	0	0	0	-4	0	0	0	-4	0	0	0
01111	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
011111	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0111111	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
01111111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
011111111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0111111111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
10	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0
001	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
0011	0	0	0	4	0	0	0	-4	0	0	0	4	0	0	0
00111	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
001111	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
0011111	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00111111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001111111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011111111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
110	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
01011	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
010111	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
0101111	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
01011111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
010111111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0101111111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
100	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
0001	0	0	0	-4	0	0	0	-4	0	0	0	-4	0	0	0
00011	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
000111	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0001111	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00011111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000111111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001111111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1110	0	0	0	-4	0	0	0	-4	0	0	0	-4	0	0	0
0110111	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
01101111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
011011111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0110111111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00101	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
001011	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0010111	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00101111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001011111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010111111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1100	0	0	0	4	0	0	0	-4	0	0	0	4	0	0	0
010011	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0100111	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
01001111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
010011111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0100111111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1000	0	0	0	-4	0	0	0	-4	0	0	0	-4	0	0	0
00001	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
000011	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
0000111	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00001111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
000011111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000111111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11110	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
011101111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0111011111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
001101	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
1									
0									
01	0								
011	0	0							
0111	0	0	0						
01111	0	0	0	0					
011111	0	0	0	0	0				
0111111	0	0	0	0	0	0			
01111111	0	0	0	0	0	0	0		
011111111	0	0	0	0	0	0	0	0	
0111111111	0	0	0	0	0	0	0	0	0
10	0								
001	0	0							
0011	0	0	0						
00111	0	0	0	0					
001111	0	0	0	0	0				
0011111	0	0	0	0	0	0			
00111111	0	0	0	0	0	0	0		
001111111	0	0	0	0	0	0	0	0	
0011111111	0	0	0	0	0	0	0	0	0
110	0	0							
01011	0	0	0	0					
010111	0	0	0	0	0				
0101111	0	0	0	0	0	0			
01011111	0	0	0	0	0	0	0		
010111111	0	0	0	0	0	0	0	0	
0101111111	0	0	0	0	0	0	0	0	0
100	0	0							
0001	0	0	0						
00011	0	0	0	0					
000111	0	0	0	0	0				
0001111	0	0	0	0	0	0			
00011111	0	0	0	0	0	0	0		
000111111	0	0	0	0	0	0	0	0	
0001111111	0	0	0	0	0	0	0	0	0
1110	0	0	0						
0110111	0	0	0	0	0	0			
01101111	0	0	0	0	0	0	0		
011011111	0	0	0	0	0	0	0	0	
0110111111	0	0	0	0	0	0	0	0	0
00101	0	0	0	0					
001011	0	0	0	0	0				
0010111	0	0	0	0	0	0			
00101111	0	0	0	0	0	0	0		
001011111	0	0	0	0	0	0	0	0	
0010111111	0	0	0	0	0	0	0	0	0
1100	0	0	0						
010011	0	0	0	0	0				
0100111	0	0	0	0	0	0			
01001111	0	0	0	0	0	0	0		
010011111	0	0	0	0	0	0	0	0	
0100111111	0	0	0	0	0	0	0	0	0
1000	0	0	0						
00001	0	0	0	0					
000011	0	0	0	0	0				
0000111	0	0	0	0	0	0			
00001111	0	0	0	0	0	0	0		
000011111	0	0	0	0	0	0	0	0	
0000111111	0	0	0	0	0	0	0	0	0
11110	0	0	0	0					
011101111	0	0	0	0	0	0	0	0	
0111011111	0	0	0	0	0	0	0	0	0
001101	0	0	0	0	0				

Self-Avoiding Minterms that Don't Contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0011011	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00110111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001101111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011011111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11010	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
0101011	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
01010111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
010101111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0101011111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000101	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
0001011	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00010111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
000101111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001011111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11100	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
01100111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
011001111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0110011111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
10100	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
0010011	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00100111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001001111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010011111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11000	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
0100011	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
01000111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
010001111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0100011111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
10000	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
000001	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0000011	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00000111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000001111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000011111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111110	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0011101	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00111011	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001110111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011101111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
01011011	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
010110111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0101101111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0001101	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00011011	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
000110111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001101111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111010	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
011010111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0110101111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0010101	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00101011	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001010111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010101111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
110010	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
01001011	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
010010111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0100101111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000101	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00001011	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000010111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000101111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111100	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
0111001111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
0011011	0	0	0	0	0	0			
00110111	0	0	0	0	0	0	0		
001101111	0	0	0	0	0	0	0	0	
0011011111	0	0	0	0	0	0	0	0	0
11010	0	0	0	0					
0101011	0	0	0	0	0	0			
01010111	0	0	0	0	0	0	0		
010101111	0	0	0	0	0	0	0	0	
0101011111	0	0	0	0	0	0	0	0	0
000101	0	0	0	0	0				
0001011	0	0	0	0	0	0			
00010111	0	0	0	0	0	0	0		
000101111	0	0	0	0	0	0	0	0	
0001011111	0	0	0	0	0	0	0	0	0
11100	0	0	0	0					
01100111	0	0	0	0	0	0	0		
011001111	0	0	0	0	0	0	0	0	
0110011111	0	0	0	0	0	0	0	0	0
10100	0	0	0	0					
0010011	0	0	0	0	0	0			
00100111	0	0	0	0	0	0	0		
001001111	0	0	0	0	0	0	0	0	
0010011111	0	0	0	0	0	0	0	0	0
11000	0	0	0	0					
0100011	0	0	0	0	0	0			
01000111	0	0	0	0	0	0	0		
010001111	0	0	0	0	0	0	0	0	
0100011111	0	0	0	0	0	0	0	0	0
10000	0	0	0	0					
000001	0	0	0	0	0				
0000011	0	0	0	0	0	0			
00000111	0	0	0	0	0	0	0		
000001111	0	0	0	0	0	0	0	0	
0000011111	0	0	0	0	0	0	0	0	0
111110	0	0	0	0	0				
0011101	0	0	0	0	0	0			
00111011	0	0	0	0	0	0	0		
001110111	0	0	0	0	0	0	0	0	
0011101111	0	0	0	0	0	0	0	0	0
01011011	0	0	0	0	0	0	0		
010110111	0	0	0	0	0	0	0	0	
0101101111	0	0	0	0	0	0	0	0	0
0001101	0	0	0	0	0	0			
00011011	0	0	0	0	0	0	0		
000110111	0	0	0	0	0	0	0	0	
0001101111	0	0	0	0	0	0	0	0	0
111010	0	0	0	0	0				
011010111	0	0	0	0	0	0	0	0	
0110101111	0	0	0	0	0	0	0	0	0
0010101	0	0	0	0	0	0			
00101011	0	0	0	0	0	0	0		
001010111	0	0	0	0	0	0	0	0	
0010101111	0	0	0	0	0	0	0	0	0
110010	0	0	0	0	0				
01001011	0	0	0	0	0	0	0		
010010111	0	0	0	0	0	0	0	0	
0100101111	0	0	0	0	0	0	0	0	0
0000101	0	0	0	0	0	0			
00001011	0	0	0	0	0	0	0		
000010111	0	0	0	0	0	0	0	0	
0000101111	0	0	0	0	0	0	0	0	0
111100	0	0	0	0	0				
0111001111	0	0	0	0	0	0	0	0	0

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
101100	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
001100111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011001111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
110100	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
01010011	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
010100111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0101001111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0001001	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00010011	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000100111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001001111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111000	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
011000111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0110001111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
101000	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
00100011	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001000111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010001111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
110000	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
01000011	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
010000111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0100001111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
100000	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0000001	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00000011	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
000000111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000001111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1111110	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00111101	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001111011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011110111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
010111011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0101110111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00011101	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
000111011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001110111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1110110	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
0110110111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00101101	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001011011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010110111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
010011011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0100110111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00001101	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000011011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000110111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1111010	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00110101	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001101011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011010111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1101010	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
010101011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0101010111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00010101	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000101011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001010111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1110010	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
0110010111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00100101	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001001011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010010111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1100010	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
010001011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
101100	0	0	0	0	0				
001100111	0	0	0	0	0	0	0	0	
0011001111	0	0	0	0	0	0	0	0	0
110100	0	0	0	0	0				
01010011	0	0	0	0	0	0	0		
010100111	0	0	0	0	0	0	0	0	
0101001111	0	0	0	0	0	0	0	0	0
0001001	0	0	0	0	0	0			
00010011	0	0	0	0	0	0	0		
000100111	0	0	0	0	0	0	0	0	
0001001111	0	0	0	0	0	0	0	0	0
111000	0	0	0	0	0				
011000111	0	0	0	0	0	0	0	0	
0110001111	0	0	0	0	0	0	0	0	0
101000	0	0	0	0	0				
00100011	0	0	0	0	0	0	0		
001000111	0	0	0	0	0	0	0	0	
0010001111	0	0	0	0	0	0	0	0	0
110000	0	0	0	0	0				
01000011	0	0	0	0	0	0	0		
010000111	0	0	0	0	0	0	0	0	
0100001111	0	0	0	0	0	0	0	0	0
100000	0	0	0	0	0				
0000001	0	0	0	0	0	0			
00000011	0	0	0	0	0	0	0		
000000111	0	0	0	0	0	0	0	0	
0000001111	0	0	0	0	0	0	0	0	0
1111110	0	0	0	0	0	0			
00111101	0	0	0	0	0	0	0		
001111011	0	0	0	0	0	0	0	0	
0011110111	0	0	0	0	0	0	0	0	0
010111011	0	0	0	0	0	0	0	0	
0101110111	0	0	0	0	0	0	0	0	0
00011101	0	0	0	0	0	0	0		
000111011	0	0	0	0	0	0	0	0	
0001110111	0	0	0	0	0	0	0	0	0
1110110	0	0	0	0	0	0			
0110110111	0	0	0	0	0	0	0	0	0
00101101	0	0	0	0	0	0	0		
001011011	0	0	0	0	0	0	0	0	
0010110111	0	0	0	0	0	0	0	0	0
010011011	0	0	0	0	0	0	0	0	
0100110111	0	0	0	0	0	0	0	0	0
00001101	0	0	0	0	0	0	0		
000011011	0	0	0	0	0	0	0	0	
0000110111	0	0	0	0	0	0	0	0	0
1111010	0	0	0	0	0	0			
00110101	0	0	0	0	0	0	0		
001101011	0	0	0	0	0	0	0	0	
0011010111	0	0	0	0	0	0	0	0	0
1101010	0	0	0	0	0	0			
010101011	0	0	0	0	0	0	0	0	
0101010111	0	0	0	0	0	0	0	0	0
00010101	0	0	0	0	0	0	0		
000101011	0	0	0	0	0	0	0	0	
0001010111	0	0	0	0	0	0	0	0	0
1110010	0	0	0	0	0	0			
0110010111	0	0	0	0	0	0	0	0	0
00100101	0	0	0	0	0	0	0		
001001011	0	0	0	0	0	0	0	0	
0010010111	0	0	0	0	0	0	0	0	0
1100010	0	0	0	0	0	0			
010001011	0	0	0	0	0	0	0	0	

Self-Avoiding Minterms that Don't Contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0100010111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00000101	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
000001011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000010111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1111100	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
1011100	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
1101100	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
010110011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0101100111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00011001	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000110011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001100111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1110100	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
0110100111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1010100	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
001010011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010100111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1100100	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
010010011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0100100111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00001001	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
000010011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000100111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1111000	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
1011000	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
0011000111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1101000	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
010100011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0101000111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1001000	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
000100011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001000111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1110000	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
0110000111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1010000	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
001000011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010000111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1100000	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
010000011	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0100000111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1000000	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00000001	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000000011	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000000111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11111110	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001111101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011111011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0101111011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000111101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001111011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
001011101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010111011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0100111011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000011101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000111011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11110110	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001101101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011011011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0101011011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000101101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001011011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11100110	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001001101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
0100010111	0	0	0	0	0	0	0	0	0
00000101	0	0	0	0	0	0	0		
000001011	0	0	0	0	0	0	0	0	
0000010111	0	0	0	0	0	0	0	0	0
1111100	0	0	0	0	0	0			
1011100	0	0	0	0	0	0			
1101100	0	0	0	0	0	0			
010110011	0	0	0	0	0	0	0	0	
0101100111	0	0	0	0	0	0	0	0	0
00011001	0	0	0	0	0	0	0		
000110011	0	0	0	0	0	0	0	0	
0001100111	0	0	0	0	0	0	0	0	0
1110100	0	0	0	0	0	0			
0110100111	0	0	0	0	0	0	0	0	0
1010100	0	0	0	0	0	0			
001010011	0	0	0	0	0	0	0	0	
0010100111	0	0	0	0	0	0	0	0	0
1100100	0	0	0	0	0	0			
010010011	0	0	0	0	0	0	0	0	
0100100111	0	0	0	0	0	0	0	0	0
00001001	0	0	0	0	0	0	0		
000010011	0	0	0	0	0	0	0	0	
0000100111	0	0	0	0	0	0	0	0	0
1111000	0	0	0	0	0	0			
1011000	0	0	0	0	0	0			
0011000111	0	0	0	0	0	0	0	0	0
1101000	0	0	0	0	0	0			
010100011	0	0	0	0	0	0	0	0	
0101000111	0	0	0	0	0	0	0	0	0
1001000	0	0	0	0	0	0			
000100011	0	0	0	0	0	0	0	0	
0001000111	0	0	0	0	0	0	0	0	0
1110000	0	0	0	0	0	0			
0110000111	0	0	0	0	0	0	0	0	0
1010000	0	0	0	0	0	0			
001000011	0	0	0	0	0	0	0	0	
0010000111	0	0	0	0	0	0	0	0	0
1100000	0	0	0	0	0	0			
010000011	0	0	0	0	0	0	0	0	
0100000111	0	0	0	0	0	0	0	0	0
1000000	0	0	0	0	0	0			
00000001	0	0	0	0	0	0	0		
000000011	0	0	0	0	0	0	0	0	
0000000111	0	0	0	0	0	0	0	0	0
11111110	0	0	0	0	0	0	0		
001111101	0	0	0	0	0	0	0	0	
0011111011	0	0	0	0	0	0	0	0	0
0101111011	0	0	0	0	0	0	0	0	0
000111101	0	0	0	0	0	0	0	0	
0001111011	0	0	0	0	0	0	0	0	0
001011101	0	0	0	0	0	0	0	0	
0010111011	0	0	0	0	0	0	0	0	0
0100111011	0	0	0	0	0	0	0	0	0
000011101	0	0	0	0	0	0	0	0	
0000111011	0	0	0	0	0	0	0	0	0
11110110	0	0	0	0	0	0	0		
001101101	0	0	0	0	0	0	0	0	
0011011011	0	0	0	0	0	0	0	0	0
0101011011	0	0	0	0	0	0	0	0	0
000101101	0	0	0	0	0	0	0	0	
0001011011	0	0	0	0	0	0	0	0	0
11100110	0	0	0	0	0	0	0		
001001101	0	0	0	0	0	0	0	0	

Self-Avoiding Minterms that Don't Contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0010011011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0100011011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000001101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000011011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11111010	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001110101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011101011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11011010	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000110101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001101011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11101010	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001010101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010101011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11001010	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0100101011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000010101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000101011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11110010	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
001100101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011001011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11010010	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0101001011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000100101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001001011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11100010	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
001000101	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010001011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11000010	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0100001011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000000101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000001011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11111100	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
10111100	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
11011100	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0101110011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000111001	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001110011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11101100	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
10101100	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0010110011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0100110011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000011001	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000110011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11110100	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
10110100	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
11010100	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0101010011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000101001	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001010011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11100100	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
10100100	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0010010011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11000100	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0100010011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000001001	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000010011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11111000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
10111000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
11011000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0101100011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
10011000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
11101000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
10101000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
0010011011	0	0	0	0	0	0	0	0	0
0100011011	0	0	0	0	0	0	0	0	0
000001101	0	0	0	0	0	0	0	0	
0000011011	0	0	0	0	0	0	0	0	0
11111010	0	0	0	0	0	0	0		
001110101	0	0	0	0	0	0	0	0	
0011101011	0	0	0	0	0	0	0	0	0
11011010	0	0	0	0	0	0	0		
000110101	0	0	0	0	0	0	0	0	
0001101011	0	0	0	0	0	0	0	0	0
11101010	0	0	0	0	0	0	0		
001010101	0	0	0	0	0	0	0	0	
0010101011	0	0	0	0	0	0	0	0	0
11001010	0	0	0	0	0	0	0		
0100101011	0	0	0	0	0	0	0	0	0
000010101	0	0	0	0	0	0	0	0	
0000101011	0	0	0	0	0	0	0	0	0
11110010	0	0	0	0	0	0	0		
001100101	0	0	0	0	0	0	0	0	
0011001011	0	0	0	0	0	0	0	0	0
11010010	0	0	0	0	0	0	0		
0101001011	0	0	0	0	0	0	0	0	0
000100101	0	0	0	0	0	0	0	0	
0001001011	0	0	0	0	0	0	0	0	0
11100010	0	0	0	0	0	0	0		
001000101	0	0	0	0	0	0	0	0	
0010001011	0	0	0	0	0	0	0	0	0
11000010	0	0	0	0	0	0	0		
0100001011	0	0	0	0	0	0	0	0	0
000000101	0	0	0	0	0	0	0	0	
0000001011	0	0	0	0	0	0	0	0	0
11111100	0	0	0	0	0	0	0		
10111100	0	0	0	0	0	0	0		
11011100	0	0	0	0	0	0	0		
0101110011	0	0	0	0	0	0	0	0	0
000111001	0	0	0	0	0	0	0	0	
0001110011	0	0	0	0	0	0	0	0	0
11101100	0	0	0	0	0	0	0		
10101100	0	0	0	0	0	0	0		
0010110011	0	0	0	0	0	0	0	0	0
0100110011	0	0	0	0	0	0	0	0	0
000011001	0	0	0	0	0	0	0	0	
0000110011	0	0	0	0	0	0	0	0	0
11110100	0	0	0	0	0	0	0		
10110100	0	0	0	0	0	0	0		
11010100	0	0	0	0	0	0	0		
0101010011	0	0	0	0	0	0	0	0	0
000101001	0	0	0	0	0	0	0	0	
0001010011	0	0	0	0	0	0	0	0	0
11100100	0	0	0	0	0	0	0		
10100100	0	0	0	0	0	0	0		
0010010011	0	0	0	0	0	0	0	0	0
11000100	0	0	0	0	0	0	0		
0100010011	0	0	0	0	0	0	0	0	0
000001001	0	0	0	0	0	0	0	0	
0000010011	0	0	0	0	0	0	0	0	0
11111000	0	0	0	0	0	0	0		
10111000	0	0	0	0	0	0	0		
11011000	0	0	0	0	0	0	0		
0101100011	0	0	0	0	0	0	0	0	0
10011000	0	0	0	0	0	0	0		
11101000	0	0	0	0	0	0	0		
10101000	0	0	0	0	0	0	0		

Self-Avoiding Minterms that Don't Contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0010100011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11001000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0100100011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000010001	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000100011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11110000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
10110000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
11010000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0101000011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
10010000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0001000011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11100000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
10100000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0010000011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11000000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0100000011	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
10000000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000000001	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000000011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111111110	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011111101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0001111101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0010111101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000111101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111101110	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011011101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0001011101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0010011101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0000011101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111110110	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011101101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0001101101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111010110	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010101101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0000101101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111100110	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011001101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0001001101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111000110	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010001101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000001101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111111010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011110101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
110111010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001110101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111011010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010110101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
110011010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000110101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111101010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011010101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
110101010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001010101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111001010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0010010101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
110001010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000010101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111110010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0011100101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
110110010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001100101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111010010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
110010010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
0010100011	0	0	0	0	0	0	0	0	0
11001000	0	0	0	0	0	0	0		
0100100011	0	0	0	0	0	0	0	0	0
000010001	0	0	0	0	0	0	0	0	
0000100011	0	0	0	0	0	0	0	0	0
11110000	0	0	0	0	0	0	0		
10110000	0	0	0	0	0	0	0		
11010000	0	0	0	0	0	0	0		
0101000011	0	0	0	0	0	0	0	0	0
10010000	0	0	0	0	0	0	0		
0001000011	0	0	0	0	0	0	0	0	0
11100000	0	0	0	0	0	0	0		
10100000	0	0	0	0	0	0	0		
0010000011	0	0	0	0	0	0	0	0	0
11000000	0	0	0	0	0	0	0		
0100000011	0	0	0	0	0	0	0	0	0
10000000	0	0	0	0	0	0	0		
000000001	0	0	0	0	0	0	0	0	
0000000011	0	0	0	0	0	0	0	0	0
111111110	0	0	0	0	0	0	0	0	
0011111101	0	0	0	0	0	0	0	0	0
0001111101	0	0	0	0	0	0	0	0	0
0010111101	0	0	0	0	0	0	0	0	0
0000111101	0	0	0	0	0	0	0	0	0
111101110	0	0	0	0	0	0	0	0	
0011011101	0	0	0	0	0	0	0	0	0
0001011101	0	0	0	0	0	0	0	0	0
0010011101	0	0	0	0	0	0	0	0	0
0000011101	0	0	0	0	0	0	0	0	0
111110110	0	0	0	0	0	0	0	0	
0011101101	0	0	0	0	0	0	0	0	0
0001101101	0	0	0	0	0	0	0	0	0
111010110	0	0	0	0	0	0	0	0	
0010101101	0	0	0	0	0	0	0	0	0
0000101101	0	0	0	0	0	0	0	0	0
111100110	0	0	0	0	0	0	0	0	
0011001101	0	0	0	0	0	0	0	0	0
0001001101	0	0	0	0	0	0	0	0	0
111000110	0	0	0	0	0	0	0	0	
0010001101	0	0	0	0	0	0	0	0	0
0000001101	0	0	0	0	0	0	0	0	0
111111010	0	0	0	0	0	0	0	0	
0011110101	0	0	0	0	0	0	0	0	0
110111010	0	0	0	0	0	0	0	0	
0001110101	0	0	0	0	0	0	0	0	0
111011010	0	0	0	0	0	0	0	0	
0010110101	0	0	0	0	0	0	0	0	0
110011010	0	0	0	0	0	0	0	0	
0000110101	0	0	0	0	0	0	0	0	0
111101010	0	0	0	0	0	0	0	0	
0011010101	0	0	0	0	0	0	0	0	0
110101010	0	0	0	0	0	0	0	0	
0001010101	0	0	0	0	0	0	0	0	0
111001010	0	0	0	0	0	0	0	0	
0010010101	0	0	0	0	0	0	0	0	0
110001010	0	0	0	0	0	0	0	0	
0000010101	0	0	0	0	0	0	0	0	0
111110010	0	0	0	0	0	0	0	0	
0011100101	0	0	0	0	0	0	0	0	0
110110010	0	0	0	0	0	0	0	0	
0001100101	0	0	0	0	0	0	0	0	0
111010010	0	0	0	0	0	0	0	0	
110010010	0	0	0	0	0	0	0	0	

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0000100101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111100010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0011000101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
110100010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001000101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111000010	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0010000101	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
110000010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000000101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111111100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
101111100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
110111100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001111001	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111011100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
101011100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000111001	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111101100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
101101100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
110101100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001011001	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111001100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
101001100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000011001	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111110100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
101110100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
110110100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0001101001	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111010100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
101010100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
110010100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000101001	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111100100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
101100100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
110100100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0001001001	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111000100	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
101000100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
110000100	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000001001	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111111000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
101111000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
110111000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
100111000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
111011000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
101011000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
110011000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000110001	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111101000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
101101000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
110101000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
100101000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
111001000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
101001000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
110001000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000010001	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111110000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
101110000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
110110000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
100110000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
111010000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
101010000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
110010000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
100010000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
0000100101	0	0	0	0	0	0	0	0	0
111100010	0	0	0	0	0	0	0	0	
0011000101	0	0	0	0	0	0	0	0	0
110100010	0	0	0	0	0	0	0	0	
0001000101	0	0	0	0	0	0	0	0	0
111000010	0	0	0	0	0	0	0	0	
0010000101	0	0	0	0	0	0	0	0	0
110000010	0	0	0	0	0	0	0	0	
0000000101	0	0	0	0	0	0	0	0	0
111111100	0	0	0	0	0	0	0	0	
101111100	0	0	0	0	0	0	0	0	
110111100	0	0	0	0	0	0	0	0	
0001111001	0	0	0	0	0	0	0	0	0
111011100	0	0	0	0	0	0	0	0	
101011100	0	0	0	0	0	0	0	0	
0000111001	0	0	0	0	0	0	0	0	0
111101100	0	0	0	0	0	0	0	0	
101101100	0	0	0	0	0	0	0	0	
110101100	0	0	0	0	0	0	0	0	
0001011001	0	0	0	0	0	0	0	0	0
111001100	0	0	0	0	0	0	0	0	
101001100	0	0	0	0	0	0	0	0	
0000011001	0	0	0	0	0	0	0	0	0
111110100	0	0	0	0	0	0	0	0	
101110100	0	0	0	0	0	0	0	0	
110110100	0	0	0	0	0	0	0	0	
0001101001	0	0	0	0	0	0	0	0	0
111010100	0	0	0	0	0	0	0	0	
101010100	0	0	0	0	0	0	0	0	
110010100	0	0	0	0	0	0	0	0	
0000101001	0	0	0	0	0	0	0	0	0
111100100	0	0	0	0	0	0	0	0	
101100100	0	0	0	0	0	0	0	0	
110100100	0	0	0	0	0	0	0	0	
0001001001	0	0	0	0	0	0	0	0	0
111000100	0	0	0	0	0	0	0	0	
101000100	0	0	0	0	0	0	0	0	
110000100	0	0	0	0	0	0	0	0	
0000001001	0	0	0	0	0	0	0	0	0
111111000	0	0	0	0	0	0	0	0	
101111000	0	0	0	0	0	0	0	0	
110111000	0	0	0	0	0	0	0	0	
100111000	0	0	0	0	0	0	0	0	
111011000	0	0	0	0	0	0	0	0	
101011000	0	0	0	0	0	0	0	0	
110011000	0	0	0	0	0	0	0	0	
0000110001	0	0	0	0	0	0	0	0	0
111101000	0	0	0	0	0	0	0	0	
101101000	0	0	0	0	0	0	0	0	
110101000	0	0	0	0	0	0	0	0	
100101000	0	0	0	0	0	0	0	0	
111001000	0	0	0	0	0	0	0	0	
101001000	0	0	0	0	0	0	0	0	
110001000	0	0	0	0	0	0	0	0	
0000010001	0	0	0	0	0	0	0	0	0
111110000	0	0	0	0	0	0	0	0	
101110000	0	0	0	0	0	0	0	0	
110110000	0	0	0	0	0	0	0	0	
100110000	0	0	0	0	0	0	0	0	
111010000	0	0	0	0	0	0	0	0	
101010000	0	0	0	0	0	0	0	0	
110010000	0	0	0	0	0	0	0	0	
100010000	0	0	0	0	0	0	0	0	

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11	1	1	-2	1	1	-2	1	1	-2	1	1	-2	1	1	-2
0*1	0	0	0	-4	0	0	0	-4	0	0	0	-4	0	0	0
0*11	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
0*111	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0*1111	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
0*11111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0*111111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0*1111111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0*11111111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00	-1	1	2	1	-1	-2	-1	1	2	1	-1	-2	-1	1	2
00*1	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
00*11	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
00*1*1	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00*111	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00*11*1	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
00*1*11	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
00*1111	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
00*1*1*1	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00*111*1	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00*11*11	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00*1*111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00*11111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00*11*1*1	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*1*11*1	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*1111*1	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*1*1*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*111*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*11*111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*1*1111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*111111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*1*1*1*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*111*1*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*11*11*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*1*111*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*11111*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*11*1*11	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*1*11*11	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*1111*11	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*1*1*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*111*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*11*1111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*1*11111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*1111111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1*0	0	0	0	-4	0	0	0	-4	0	0	0	-4	0	0	0
0*0*11	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
0*0*111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0*0*1111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0*0*11*11	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0*0*11111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0*0*111*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0*0*11*111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0*0*111111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*1	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
000*11	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
000*1*1	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000*111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000*11*1	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
000*1*11	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
000*1111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
000*1*1*1	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000*111*1	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000*11*11	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000*1*111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
11	1								
0*1	1	0							
0*11	1	0	0						
0*111	1	0	0	0					
0*1111	1	0	0	0	0				
0*11111	1	0	0	0	0	0			
0*111111	1	0	0	0	0	0	0		
0*1111111	1	0	0	0	0	0	0	0	
0*11111111	1	0	0	0	0	0	0	0	0
00	1								
00*1	1	0	0						
00*11	1	0	0	0					
00*1*1	1	0	0	0	0				
00*111	1	0	0	0	0				
00*11*1	1	0	0	0	0	0			
00*1*11	1	0	0	0	0	0			
00*1111	1	0	0	0	0	0			
00*1*1*1	1	0	0	0	0	0	0		
00*111*1	1	0	0	0	0	0	0		
00*11*11	1	0	0	0	0	0	0		
00*1*111	1	0	0	0	0	0	0		
00*11111	1	0	0	0	0	0	0		
00*11*1*1	1	0	0	0	0	0	0	0	
00*1*11*1	1	0	0	0	0	0	0	0	
00*1111*1	1	0	0	0	0	0	0	0	
00*1*1*11	1	0	0	0	0	0	0	0	
00*111*11	1	0	0	0	0	0	0	0	
00*11*111	1	0	0	0	0	0	0	0	
00*1*1111	1	0	0	0	0	0	0	0	
00*111111	1	0	0	0	0	0	0	0	
00*1*1*1*1	1	0	0	0	0	0	0	0	0
00*111*1*1	1	0	0	0	0	0	0	0	0
00*11*11*1	1	0	0	0	0	0	0	0	0
00*1*111*1	1	0	0	0	0	0	0	0	0
00*11111*1	1	0	0	0	0	0	0	0	0
00*11*1*11	1	0	0	0	0	0	0	0	0
00*1*11*11	1	0	0	0	0	0	0	0	0
00*1111*11	1	0	0	0	0	0	0	0	0
00*1*1*111	1	0	0	0	0	0	0	0	0
00*111*111	1	0	0	0	0	0	0	0	0
00*11*1111	1	0	0	0	0	0	0	0	0
00*1*11111	1	0	0	0	0	0	0	0	0
00*1111111	1	0	0	0	0	0	0	0	0
1*0	1	0							
0*0*11	1	0	0	0	0				
0*0*111	1	0	0	0	0	0			
0*0*1111	1	0	0	0	0	0	0		
0*0*11*11	1	0	0	0	0	0	0	0	
0*0*11111	1	0	0	0	0	0	0	0	
0*0*111*11	1	0	0	0	0	0	0	0	0
0*0*11*111	1	0	0	0	0	0	0	0	0
0*0*111111	1	0	0	0	0	0	0	0	0
000*1	1	0	0	0					
000*11	1	0	0	0	0				
000*1*1	1	0	0	0	0	0			
000*111	1	0	0	0	0	0			
000*11*1	1	0	0	0	0	0	0		
000*1*11	1	0	0	0	0	0	0		
000*1111	1	0	0	0	0	0	0		
000*1*1*1	1	0	0	0	0	0	0	0	
000*111*1	1	0	0	0	0	0	0	0	
000*11*11	1	0	0	0	0	0	0	0	
000*1*111	1	0	0	0	0	0	0	0	

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
000*11111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000*11*1*1	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*1*11*1	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*1111*1	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*1*1*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*111*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*11*111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*1*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*111111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*0	0	0	0	0	-5	0	0	0	0	-5	0	0	0	0	-5
00*0*11	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
00*0*111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00*0*11*1	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*0*1111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*0*111*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*0*11*11	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*0*11111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1*00	0	0	0	0	5	0	0	0	0	-5	0	0	0	0	5
0*00*11	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0*00*111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0*00*1*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0*00*1111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0*00*11*11	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0*00*1*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0*00*11111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000*1	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
0000*11	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
0000*1*1	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000*111	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000*11*1	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000*1*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000*1111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000*1*1*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000*111*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000*11*11	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000*1*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000*11111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
111*0	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0*1*0*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0*0*0*11	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0*0*0*111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0*0*0*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*0*11	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
000*0*111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
000*0*11*1	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*0*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*00	0	0	0	0	0	6	0	0	0	0	0	-6	0	0	0
00*00*11	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
00*00*1*1	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00*00*111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00*00*11*1	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00*00*1*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00*00*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1*000	0	0	0	0	0	-6	0	0	0	0	0	-6	0	0	0
0*000*11	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0*000*111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0*000*1*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0*000*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00000*1	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
00000*11	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
00000*1*1	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00000*111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00000*11*1	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
000*11111	1	0	0	0	0	0	0	0	
000*11*1*1	1	0	0	0	0	0	0	0	0
000*1*11*1	1	0	0	0	0	0	0	0	0
000*1111*1	1	0	0	0	0	0	0	0	0
000*1*1*11	1	0	0	0	0	0	0	0	0
000*111*11	1	0	0	0	0	0	0	0	0
000*11*111	1	0	0	0	0	0	0	0	0
000*1*1111	1	0	0	0	0	0	0	0	0
000*111111	1	0	0	0	0	0	0	0	0
11*0	1	0	0						
00*0*11	1	0	0	0	0	0			
00*0*111	1	0	0	0	0	0	0		
00*0*11*1	1	0	0	0	0	0	0	0	
00*0*1111	1	0	0	0	0	0	0	0	
00*0*111*1	1	0	0	0	0	0	0	0	0
00*0*11*11	1	0	0	0	0	0	0	0	0
00*0*11111	1	0	0	0	0	0	0	0	0
1*00	1	0	0						
0*00*11	1	0	0	0	0	0			
0*00*111	1	0	0	0	0	0	0		
0*00*1*11	1	0	0	0	0	0	0	0	
0*00*1111	1	0	0	0	0	0	0	0	
0*00*11*11	1	0	0	0	0	0	0	0	0
0*00*1*111	1	0	0	0	0	0	0	0	0
0*00*11111	1	0	0	0	0	0	0	0	0
0000*1	1	0	0	0	0				
0000*11	1	0	0	0	0	0			
0000*1*1	1	0	0	0	0	0	0		
0000*111	1	0	0	0	0	0	0		
0000*11*1	1	0	0	0	0	0	0	0	
0000*1*11	1	0	0	0	0	0	0	0	
0000*1111	1	0	0	0	0	0	0	0	
0000*1*1*1	1	0	0	0	0	0	0	0	0
0000*111*1	1	0	0	0	0	0	0	0	0
0000*11*11	1	0	0	0	0	0	0	0	0
0000*1*111	1	0	0	0	0	0	0	0	0
0000*11111	1	0	0	0	0	0	0	0	0
111*0	1	0	0	0					
0*1*0*1111	1	0	0	0	0	0	0	0	0
0*0*0*11	1	0	0	0	0	0	0		
0*0*0*111	1	0	0	0	0	0	0	0	
0*0*0*1111	1	0	0	0	0	0	0	0	0
000*0*11	1	0	0	0	0	0	0		
000*0*111	1	0	0	0	0	0	0	0	
000*0*11*1	1	0	0	0	0	0	0	0	0
000*0*1111	1	0	0	0	0	0	0	0	0
11*00	1	0	0	0					
00*00*11	1	0	0	0	0	0	0		
00*00*1*1	1	0	0	0	0	0	0	0	
00*00*111	1	0	0	0	0	0	0	0	
00*00*11*1	1	0	0	0	0	0	0	0	0
00*00*1*11	1	0	0	0	0	0	0	0	0
00*00*1111	1	0	0	0	0	0	0	0	0
1*000	1	0	0	0					
0*000*11	1	0	0	0	0	0	0		
0*000*111	1	0	0	0	0	0	0	0	
0*000*1*11	1	0	0	0	0	0	0	0	0
0*000*1111	1	0	0	0	0	0	0	0	0
00000*1	1	0	0	0	0	0			
00000*11	1	0	0	0	0	0	0		
00000*1*1	1	0	0	0	0	0	0	0	
00000*111	1	0	0	0	0	0	0	0	
00000*11*1	1	0	0	0	0	0	0	0	0

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00000*1*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00000*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1111*0	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
11*0*0	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00*0*0*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*0*0*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0*00*0*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0*00*0*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000*0*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000*0*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1*1*00	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
111*00	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
0*0*00*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0*0*00*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000*00*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000*00*1*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000*00*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11*000	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00*000*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*000*1*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*000*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1*0000	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
0*0000*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0*0000*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000000*1	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
000000*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000000*1*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000000*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11111*0	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
111*0*0	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0*0*0*0*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*0*0*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*00*0	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
00*00*0*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0*000*0*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00000*0*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*1*00	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
1*11*00	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
1111*00	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
11*0*00	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
00*0*00*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0*00*00*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0000*00*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1*1*000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
111*000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0*0*000*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*000*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*0000	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0
00*0000*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1*00000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
0*00000*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0000000*1	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
0000000*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
111111*0	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
0000*1*0*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1111*0*0	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
11*0*0*0	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
111*00*0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
11*000*0	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
1*1*1*00	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
111*1*00	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
11*11*00	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
1*111*00	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
00000*1*11	1	0	0	0	0	0	0	0	0
00000*1111	1	0	0	0	0	0	0	0	0
1111*0	1	0	0	0	0				
11*0*0	1	0	0	0	0				
00*0*0*11	1	0	0	0	0	0	0	0	
00*0*0*111	1	0	0	0	0	0	0	0	0
0*00*0*11	1	0	0	0	0	0	0	0	
0*00*0*111	1	0	0	0	0	0	0	0	0
0000*0*11	1	0	0	0	0	0	0	0	
0000*0*111	1	0	0	0	0	0	0	0	0
1*1*00	1	0	0	0	0				
111*00	1	0	0	0	0				
0*0*00*11	1	0	0	0	0	0	0	0	
0*0*00*111	1	0	0	0	0	0	0	0	0
000*00*11	1	0	0	0	0	0	0	0	
000*00*1*1	1	0	0	0	0	0	0	0	0
000*00*111	1	0	0	0	0	0	0	0	0
11*000	1	0	0	0	0				
00*000*11	1	0	0	0	0	0	0	0	
00*000*1*1	1	0	0	0	0	0	0	0	0
00*000*111	1	0	0	0	0	0	0	0	0
1*0000	1	0	0	0	0				
0*0000*11	1	0	0	0	0	0	0	0	
0*0000*111	1	0	0	0	0	0	0	0	0
000000*1	1	0	0	0	0	0	0		
000000*11	1	0	0	0	0	0	0	0	
000000*1*1	1	0	0	0	0	0	0	0	0
000000*111	1	0	0	0	0	0	0	0	0
11111*0	1	0	0	0	0	0			
111*0*0	1	0	0	0	0	0			
0*0*0*0*11	1	0	0	0	0	0	0	0	0
000*0*0*11	1	0	0	0	0	0	0	0	0
11*00*0	1	0	0	0	0	0			
00*00*0*11	1	0	0	0	0	0	0	0	0
0*000*0*11	1	0	0	0	0	0	0	0	0
00000*0*11	1	0	0	0	0	0	0	0	0
11*1*00	1	0	0	0	0	0			
1*11*00	1	0	0	0	0	0			
1111*00	1	0	0	0	0	0			
11*0*00	1	0	0	0	0	0			
00*0*00*11	1	0	0	0	0	0	0	0	0
0*00*00*11	1	0	0	0	0	0	0	0	0
0000*00*11	1	0	0	0	0	0	0	0	0
1*1*000	1	0	0	0	0	0			
111*000	1	0	0	0	0	0			
0*0*000*11	1	0	0	0	0	0	0	0	0
000*000*11	1	0	0	0	0	0	0	0	0
11*0000	1	0	0	0	0	0			
00*0000*11	1	0	0	0	0	0	0	0	0
1*00000	1	0	0	0	0	0			
0*00000*11	1	0	0	0	0	0	0	0	0
0000000*1	1	0	0	0	0	0	0	0	
0000000*11	1	0	0	0	0	0	0	0	0
111111*0	1	0	0	0	0	0	0		
0000*1*0*1	1	0	0	0	0	0	0	0	0
1111*0*0	1	0	0	0	0	0	0		
11*0*0*0	1	0	0	0	0	0	0		
111*00*0	1	0	0	0	0	0	0		
11*000*0	1	0	0	0	0	0	0		
1*1*1*00	1	0	0	0	0	0	0		
111*1*00	1	0	0	0	0	0	0		
11*11*00	1	0	0	0	0	0	0		
1*111*00	1	0	0	0	0	0	0		

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11111*00	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
111*0*00	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
11*00*00	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
11*1*000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
1*11*000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
1111*000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
11*0*000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
1*1*0000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
111*0000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
11*00000	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
1*000000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00000000*1	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1111111*0	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11*11*0*0	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11111*0*0	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111*0*0*0	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11*00*0*0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*1*00*0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1111*00*0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*0*00*0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111*000*0	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11*0000*0	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*1*1*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*11*1*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1111*1*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*1*11*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111*11*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*111*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*1111*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111111*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*11*0*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1111*0*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*0*0*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*1*00*00	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111*00*00	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11*000*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*1*1*000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111*1*000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11*11*000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1*111*000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11111*000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111*0*000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11*00*000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*1*0000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*11*0000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1111*0000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*0*0000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*1*00000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111*00000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11*000000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*0000000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
11111111*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1111*0*1*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
111*11*0*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*111*0*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
111111*0*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1111*0*0*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*0*0*0*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
111*00*0*0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11*000*0*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
111*1*00*0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11*11*00*0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11111*00*0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
11111*00	1	0	0	0	0	0	0		
111*0*00	1	0	0	0	0	0	0		
11*00*00	1	0	0	0	0	0	0		
11*1*000	1	0	0	0	0	0	0		
1*11*000	1	0	0	0	0	0	0		
1111*000	1	0	0	0	0	0	0		
11*0*000	1	0	0	0	0	0	0		
1*1*0000	1	0	0	0	0	0	0		
111*0000	1	0	0	0	0	0	0		
11*00000	1	0	0	0	0	0	0		
1*000000	1	0	0	0	0	0	0		
00000000*1	1	0	0	0	0	0	0	0	0
1111111*0	1	0	0	0	0	0	0	0	
11*11*0*0	1	0	0	0	0	0	0	0	
11111*0*0	1	0	0	0	0	0	0	0	
111*0*0*0	1	0	0	0	0	0	0	0	
11*00*0*0	1	0	0	0	0	0	0	0	
11*1*00*0	1	0	0	0	0	0	0	0	
1111*00*0	1	0	0	0	0	0	0	0	
11*0*00*0	1	0	0	0	0	0	0	0	
111*000*0	1	0	0	0	0	0	0	0	
11*0000*0	1	0	0	0	0	0	0	0	
11*1*1*00	1	0	0	0	0	0	0	0	
1*11*1*00	1	0	0	0	0	0	0	0	
1111*1*00	1	0	0	0	0	0	0	0	
1*1*11*00	1	0	0	0	0	0	0	0	
111*11*00	1	0	0	0	0	0	0	0	
11*111*00	1	0	0	0	0	0	0	0	
1*1111*00	1	0	0	0	0	0	0	0	
111111*00	1	0	0	0	0	0	0	0	
1*11*0*00	1	0	0	0	0	0	0	0	
1111*0*00	1	0	0	0	0	0	0	0	
11*0*0*00	1	0	0	0	0	0	0	0	
1*1*00*00	1	0	0	0	0	0	0	0	
111*00*00	1	0	0	0	0	0	0	0	
11*000*00	1	0	0	0	0	0	0	0	
1*1*1*000	1	0	0	0	0	0	0	0	
111*1*000	1	0	0	0	0	0	0	0	
11*11*000	1	0	0	0	0	0	0	0	
1*111*000	1	0	0	0	0	0	0	0	
11111*000	1	0	0	0	0	0	0	0	
111*0*000	1	0	0	0	0	0	0	0	
11*00*000	1	0	0	0	0	0	0	0	
11*1*0000	1	0	0	0	0	0	0	0	
1*11*0000	1	0	0	0	0	0	0	0	
1111*0000	1	0	0	0	0	0	0	0	
11*0*0000	1	0	0	0	0	0	0	0	
1*1*00000	1	0	0	0	0	0	0	0	
111*00000	1	0	0	0	0	0	0	0	
11*000000	1	0	0	0	0	0	0	0	
1*0000000	1	0	0	0	0	0	0	0	
11111111*0	1	0	0	0	0	0	0	0	0
1111*0*1*0	1	0	0	0	0	0	0	0	0
111*11*0*0	1	0	0	0	0	0	0	0	0
11*111*0*0	1	0	0	0	0	0	0	0	0
111111*0*0	1	0	0	0	0	0	0	0	0
1111*0*0*0	1	0	0	0	0	0	0	0	0
11*0*0*0*0	1	0	0	0	0	0	0	0	0
111*00*0*0	1	0	0	0	0	0	0	0	0
11*000*0*0	1	0	0	0	0	0	0	0	0
111*1*00*0	1	0	0	0	0	0	0	0	0
11*11*00*0	1	0	0	0	0	0	0	0	0
11111*00*0	1	0	0	0	0	0	0	0	0

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
111*0*00*0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11*00*00*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*1*000*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1111*000*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*0*000*0	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
01*1	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0
01*11	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
01*111	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
01*1111	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
01*11111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
01*111111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
01*1111111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
101	0	-2	-3	-2	-5	-5	-7	-10	-12	-17	-22	-29	-39	-51	-68
010	0	-2	3	-2	5	-5	7	-10	12	-17	22	-29	39	-51	68
0*01	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0
0101	0	-2	0	2	0	4	0	2	0	-2	0	-4	0	-2	0
010**11	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0101*11	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0101**11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
010**111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0101*111	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0101**111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
010**1111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0101*1111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
010**11*11	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
0101*11*11	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
0101**1111	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
010**11111	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
0101*11111	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
1*10	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0
10*0	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0	-2	0
1010	0	-2	0	2	0	4	0	2	0	-2	0	-4	0	-2	0
00*01	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
00*01*11	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00*01**11	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*01*111	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
00*01**111	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
00*01*1111	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
11*10	0	0	0	0	0	0	-7	0	0	0	0	0	0	-7	0
00**01*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00**01*111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
01010**11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
010*01*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
010101*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
010*01**11	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
010101**11	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
01010**111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
010*01*111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
010101*111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000*01	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
000*01*11	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000*01**11	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000*01*111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
10*00	0	0	0	0	0	0	7	0	0	0	0	0	0	-7	0
111*10	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
00**101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
00**101*11	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000**01*11	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
00*0101	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
00*010**11	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
00*0101*11	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
0000*01	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
0000*01*11	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
111*0*00*0	1	0	0	0	0	0	0	0	0
11*00*00*0	1	0	0	0	0	0	0	0	0
11*1*000*0	1	0	0	0	0	0	0	0	0
1111*000*0	1	0	0	0	0	0	0	0	0
11*0*000*0	1	0	0	0	0	0	0	0	0
01*1	0	1	0						
01*11	0	1	0	0					
01*111	0	1	0	0	0				
01*1111	0	1	0	0	0	0			
01*11111	0	1	0	0	0	0	0		
01*111111	0	1	0	0	0	0	0	0	
01*1111111	0	1	0	0	0	0	0	0	0
101	0	1							
010	0	1							
0*01	0	1	0						
0101	0	1	0						
010**11	0	1	0	0	0	0			
0101*11	0	1	0	0	0	0			
0101**11	0	1	0	0	0	0	0		
010**111	0	1	0	0	0	0	0		
0101*111	0	1	0	0	0	0	0		
0101**111	0	1	0	0	0	0	0	0	
010**1111	0	1	0	0	0	0	0	0	
0101*1111	0	1	0	0	0	0	0	0	
010**11*11	0	1	0	0	0	0	0	0	0
0101*11*11	0	1	0	0	0	0	0	0	0
0101**1111	0	1	0	0	0	0	0	0	0
010**11111	0	1	0	0	0	0	0	0	0
0101*11111	0	1	0	0	0	0	0	0	0
1*10	0	1	0						
10*0	0	1	0						
1010	0	1	0						
00*01	0	1	0	0					
00*01*11	0	1	0	0	0	0	0		
00*01**11	0	1	0	0	0	0	0	0	
00*01*111	0	1	0	0	0	0	0	0	
00*01**111	0	1	0	0	0	0	0	0	0
00*01*1111	0	1	0	0	0	0	0	0	0
11*10	0	1	0	0					
00**01*11	0	1	0	0	0	0	0	0	
00**01*111	0	1	0	0	0	0	0	0	0
01010**11	0	1	0	0	0	0	0	0	
010*01*11	0	1	0	0	0	0	0	0	
010101*11	0	1	0	0	0	0	0	0	
010*01**11	0	1	0	0	0	0	0	0	0
010101**11	0	1	0	0	0	0	0	0	0
01010**111	0	1	0	0	0	0	0	0	0
010*01*111	0	1	0	0	0	0	0	0	0
010101*111	0	1	0	0	0	0	0	0	0
000*01	0	1	0	0	0				
000*01*11	0	1	0	0	0	0	0	0	
000*01**11	0	1	0	0	0	0	0	0	0
000*01*111	0	1	0	0	0	0	0	0	0
10*00	0	1	0	0					
111*10	0	1	0	0	0				
00**101	0	1	0	0	0	0			
00**101*11	0	1	0	0	0	0	0	0	0
000**01*11	0	1	0	0	0	0	0	0	0
00*0101	0	1	0	0	0	0			
00*010**11	0	1	0	0	0	0	0	0	0
00*0101*11	0	1	0	0	0	0	0	0	0
0000*01	0	1	0	0	0	0			
0000*01*11	0	1	0	0	0	0	0	0	0

Minterms with One Cyclic Shift and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
010*00**11	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
10*000	0	0	0	0	0	0	0	-8	0	0	0	0	0	0	0
1111*10	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
000**101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11**010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
11*1010	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
00**0101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
000*0101	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00000*01	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
101**00	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
1010*00	0	0	0	0	0	0	0	0	-9	0	0	0	0	0	0
10*0000	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0
11111*10	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
00*01*101	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000**101	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
111**010	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11**1010	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
111*1010	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
00**10101	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000**0101	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00*010101	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
0000*0101	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
000000*01	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1010**00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
11*10*00	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
101**000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1010*000	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
10*00000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111111*10	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00**11*101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
00**01*101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000*01*101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
00*00**101	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
00000**101	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
1111**010	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
111**1010	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1111*1010	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000**10101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
0000**0101	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
11*10*010	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11**01010	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*101010	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00**010101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000*010101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
00*00*0101	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
00000*0101	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
0000000*01	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
10101**00	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11*10**00	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11**10*00	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
111*10*00	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
101*10*00	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
101010*00	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1010**000	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
11*10*000	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
101**0000	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
1010*0000	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
10*000000	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1111111*10	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
11*11**010	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
11111**010	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
1111**1010	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
11*11*1010	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
010*00**11	0	1	0	0	0	0	0	0	0
10*000	0	1	0	0	0				
1111*10	0	1	0	0	0	0			
000**101	0	1	0	0	0	0	0		
11**010	0	1	0	0	0	0			
11*1010	0	1	0	0	0	0			
00**0101	0	1	0	0	0	0	0		
000*0101	0	1	0	0	0	0	0		
00000*01	0	1	0	0	0	0	0		
101**00	0	1	0	0	0	0			
1010*00	0	1	0	0	0	0			
10*0000	0	1	0	0	0	0			
11111*10	0	1	0	0	0	0	0		
00*01*101	0	1	0	0	0	0	0	0	
0000**101	0	1	0	0	0	0	0	0	
111**010	0	1	0	0	0	0	0		
11**1010	0	1	0	0	0	0	0		
111*1010	0	1	0	0	0	0	0		
00**10101	0	1	0	0	0	0	0	0	
000**0101	0	1	0	0	0	0	0	0	
00*010101	0	1	0	0	0	0	0	0	
0000*0101	0	1	0	0	0	0	0	0	
000000*01	0	1	0	0	0	0	0	0	
1010**00	0	1	0	0	0	0	0		
11*10*00	0	1	0	0	0	0	0		
101**000	0	1	0	0	0	0	0		
1010*000	0	1	0	0	0	0	0		
10*00000	0	1	0	0	0	0	0		
111111*10	0	1	0	0	0	0	0	0	
00**11*101	0	1	0	0	0	0	0	0	0
00**01*101	0	1	0	0	0	0	0	0	0
000*01*101	0	1	0	0	0	0	0	0	0
00*00**101	0	1	0	0	0	0	0	0	0
00000**101	0	1	0	0	0	0	0	0	0
1111**010	0	1	0	0	0	0	0	0	
111**1010	0	1	0	0	0	0	0	0	
1111*1010	0	1	0	0	0	0	0	0	
000**10101	0	1	0	0	0	0	0	0	0
0000**0101	0	1	0	0	0	0	0	0	0
11*10*010	0	1	0	0	0	0	0	0	
11**01010	0	1	0	0	0	0	0	0	
11*101010	0	1	0	0	0	0	0	0	
00**010101	0	1	0	0	0	0	0	0	0
000*010101	0	1	0	0	0	0	0	0	0
00*00*0101	0	1	0	0	0	0	0	0	0
00000*0101	0	1	0	0	0	0	0	0	0
0000000*01	0	1	0	0	0	0	0	0	0
10101**00	0	1	0	0	0	0	0	0	
11*10**00	0	1	0	0	0	0	0	0	
11**10*00	0	1	0	0	0	0	0	0	
111*10*00	0	1	0	0	0	0	0	0	
101*10*00	0	1	0	0	0	0	0	0	
101010*00	0	1	0	0	0	0	0	0	
1010**000	0	1	0	0	0	0	0	0	
11*10*000	0	1	0	0	0	0	0	0	
101**0000	0	1	0	0	0	0	0	0	
1010*0000	0	1	0	0	0	0	0	0	
10*000000	0	1	0	0	0	0	0	0	
1111111*10	0	1	0	0	0	0	0	0	0
11*11**010	0	1	0	0	0	0	0	0	0
11111**010	0	1	0	0	0	0	0	0	0
1111**1010	0	1	0	0	0	0	0	0	0
11*11*1010	0	1	0	0	0	0	0	0	0

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
11111*1010	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
111**010*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*1010*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11**10*010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
111*10*010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
111**01010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
11**101010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
111*101010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
11**00*010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
011*1	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
011*11	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
011*111	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
011*1111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
011*11111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
011*111111	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
1011	0	0	-3	-4	0	-3	-7	-4	-3	-10	-11	-7	-13	-21	-18
001**1	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
001**111	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
001**1111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
001**11111	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
1101	0	0	-3	-4	0	-3	-7	-4	-3	-10	-11	-7	-13	-21	-18
010*1	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
010*1*111	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
010*1*1111	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
1001	0	0	3	4	0	-3	-7	-4	3	10	11	1	-13	-21	-12
0110	0	0	-3	4	0	-3	7	-4	-3	10	-11	1	13	-21	12
0*101	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
01101	0	0	-3	0	5	-3	0	8	-3	-5	11	-3	-13	14	2
0**011	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
01*011	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
0*1011	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
011011	0	0	-3	0	0	3	0	0	6	0	0	3	0	0	-3
0110***111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
01101**111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
0110*1*111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
011011*111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
0010	0	0	3	-4	0	-3	7	-4	3	-10	11	-7	13	-21	18
0010*1	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
0010***111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
0010*1*111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
0100	0	0	3	-4	0	-3	7	-4	3	-10	11	-7	13	-21	18
0*001	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
01001	0	0	3	0	-5	-3	0	8	3	-5	-11	-3	13	14	-2
0100***111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
01001**111	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
1*110	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
101*0	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
10110	0	0	-3	0	5	-3	0	8	-3	-5	11	-3	-13	14	2
001*01	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
001*011	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
1*010	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
100*0	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
10010	0	0	3	0	-5	-3	0	8	3	-5	-11	-3	13	14	-2
00*001	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
001001	0	0	3	0	0	3	0	0	-6	0	0	3	0	0	3
11*110	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
110**0	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
1101*0	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
110*10	0	0	-3	0	0	-3	0	0	-3	0	0	-3	0	0	-3
110110	0	0	-3	0	0	3	0	0	6	0	0	3	0	0	-3
010**011	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
010*1011	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000**011	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
11111*1010	0	1	0	0	0	0	0	0	0
111**010*0	0	1	0	0	0	0	0	0	0
111*1010*0	0	1	0	0	0	0	0	0	0
11**10*010	0	1	0	0	0	0	0	0	0
111*10*010	0	1	0	0	0	0	0	0	0
111**01010	0	1	0	0	0	0	0	0	0
11**101010	0	1	0	0	0	0	0	0	0
111*101010	0	1	0	0	0	0	0	0	0
11**00*010	0	1	0	0	0	0	0	0	0
011*1	0	0	1	0					
011*11	0	0	1	0	0				
011*111	0	0	1	0	0	0			
011*1111	0	0	1	0	0	0	0		
011*11111	0	0	1	0	0	0	0	0	
011*111111	0	0	1	0	0	0	0	0	0
1011	0	0	1						
001**1	0	0	1	0	0				
001**111	0	0	1	0	0	0	0		
001**1111	0	0	1	0	0	0	0	0	
001**11111	0	0	1	0	0	0	0	0	0
1101	0	0	1						
010*1	0	0	1	0					
010*1*111	0	0	1	0	0	0	0	0	
010*1*1111	0	0	1	0	0	0	0	0	0
1001	0	0	1						
0110	0	0	1						
0*101	0	0	1	0					
01101	0	0	1	0					
0**011	0	0	1	0	0				
01*011	0	0	1	0	0				
0*1011	0	0	1	0	0				
011011	0	0	1	0	0				
0110***111	0	0	1	0	0	0	0	0	0
01101**111	0	0	1	0	0	0	0	0	0
0110*1*111	0	0	1	0	0	0	0	0	0
011011*111	0	0	1	0	0	0	0	0	0
0010	0	0	1						
0010*1	0	0	1	0	0				
0010***111	0	0	1	0	0	0	0	0	0
0010*1*111	0	0	1	0	0	0	0	0	0
0100	0	0	1						
0*001	0	0	1	0					
01001	0	0	1	0					
0100***111	0	0	1	0	0	0	0	0	0
01001**111	0	0	1	0	0	0	0	0	0
1*110	0	0	1	0					
101*0	0	0	1	0					
10110	0	0	1	0					
001*01	0	0	1	0	0				
001*011	0	0	1	0	0	0			
1*010	0	0	1	0					
100*0	0	0	1	0					
10010	0	0	1	0					
00*001	0	0	1	0	0				
001001	0	0	1	0	0				
11*110	0	0	1	0	0				
110**0	0	0	1	0	0				
1101*0	0	0	1	0	0				
110*10	0	0	1	0	0				
110110	0	0	1	0	0				
010**011	0	0	1	0	0	0	0		
010*1011	0	0	1	0	0	0	0		
000**011	0	0	1	0	0	0	0		

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1**100	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
10*100	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
1*0100	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
100*00	0	0	3	0	0	-3	0	0	3	0	0	-3	0	0	3
100100	0	0	3	0	0	3	0	0	-6	0	0	3	0	0	3
000*001	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
111*110	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
001**101	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
001**1011	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
010*1*011	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
0010*101	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
0010**011	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
0010*1011	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
0100**011	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
01001*011	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
0000**011	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
110*100	0	0	0	0	0	0	0	0	0	10	0	0	0	0	0
0000*001	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
100*000	0	0	0	0	0	0	0	0	0	-10	0	0	0	0	0
1111*110	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
001**1101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
001**11011	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
000***1011	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
0010*1*011	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
001*0*101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
001*01101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
001*0*1011	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
001*011011	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
000*0*101	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000*0*1011	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
00100**011	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
001001*011	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
00000**011	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
110**010	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
1101*010	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
000**010*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*0010*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111**100	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
101**100	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
101*0*00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101*0100	0	0	0	0	0	0	0	0	0	0	-11	0	0	0	0
00000*001	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
110**000	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
100*0000	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0
11111*110	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000***1101	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
000*0*1*01	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000**01101	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
0000*0*101	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
000*001*01	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*1*010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
110*1*0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110*1*010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
110*100*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110**0010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
110*10010	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
1111**100	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
1011***00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1011**100	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
1101***00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1101**100	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000***1001	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
10110**00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
1**100	0	0	1	0	0				
10*100	0	0	1	0	0				
1*0100	0	0	1	0	0				
100*00	0	0	1	0	0				
100100	0	0	1	0	0				
000*001	0	0	1	0	0	0			
111*110	0	0	1	0	0	0			
001**101	0	0	1	0	0	0	0		
001**1011	0	0	1	0	0	0	0	0	
010*1*011	0	0	1	0	0	0	0	0	
0010*101	0	0	1	0	0	0	0		
0010**011	0	0	1	0	0	0	0	0	
0010*1011	0	0	1	0	0	0	0	0	
0100**011	0	0	1	0	0	0	0	0	
01001*011	0	0	1	0	0	0	0	0	
0000**011	0	0	1	0	0	0	0	0	
110*100	0	0	1	0	0	0			
0000*001	0	0	1	0	0	0	0		
100*000	0	0	1	0	0	0			
1111*110	0	0	1	0	0	0	0		
001**1101	0	0	1	0	0	0	0	0	
001**11011	0	0	1	0	0	0	0	0	0
000***1011	0	0	1	0	0	0	0	0	0
0010*1*011	0	0	1	0	0	0	0	0	0
001*0*101	0	0	1	0	0	0	0	0	
001*01101	0	0	1	0	0	0	0	0	
001*0*1011	0	0	1	0	0	0	0	0	0
001*011011	0	0	1	0	0	0	0	0	0
000*0*101	0	0	1	0	0	0	0	0	
000*0*1011	0	0	1	0	0	0	0	0	0
00100**011	0	0	1	0	0	0	0	0	0
001001*011	0	0	1	0	0	0	0	0	0
00000**011	0	0	1	0	0	0	0	0	0
110**010	0	0	1	0	0	0	0		
1101*010	0	0	1	0	0	0	0		
000**010*1	0	0	1	0	0	0	0	0	0
000*0010*1	0	0	1	0	0	0	0	0	0
111**100	0	0	1	0	0	0	0		
101**100	0	0	1	0	0	0	0		
101*0*00	0	0	1	0	0	0	0		
101*0100	0	0	1	0	0	0	0		
00000*001	0	0	1	0	0	0	0	0	
110**000	0	0	1	0	0	0	0		
100*0000	0	0	1	0	0	0	0		
11111*110	0	0	1	0	0	0	0	0	
000***1101	0	0	1	0	0	0	0	0	0
000*0*1*01	0	0	1	0	0	0	0	0	0
000**01101	0	0	1	0	0	0	0	0	0
0000*0*101	0	0	1	0	0	0	0	0	0
000*001*01	0	0	1	0	0	0	0	0	0
111*1*010	0	0	1	0	0	0	0	0	
110*1*0*0	0	0	1	0	0	0	0	0	
110*1*010	0	0	1	0	0	0	0	0	
110*100*0	0	0	1	0	0	0	0	0	
110**0010	0	0	1	0	0	0	0	0	
110*10010	0	0	1	0	0	0	0	0	
1111**100	0	0	1	0	0	0	0	0	
1011***00	0	0	1	0	0	0	0	0	
1011**100	0	0	1	0	0	0	0	0	
1101***00	0	0	1	0	0	0	0	0	
1101**100	0	0	1	0	0	0	0	0	
000***1001	0	0	1	0	0	0	0	0	0
10110**00	0	0	1	0	0	0	0	0	

Minterms with One Cyclic Shift
and whose related minterm doesn't contain *

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
101*0*100	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
10110*100	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000*0*1001	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
1101*0*00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110**0100	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
1101*0100	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
000**01001	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
000*001001	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
000000*001	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
101*0*000	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
110**0000	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
100*00000	0	0	0	0	0	0	0	0	0	0	0	-12	0	0	0
111111*110	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
111*110**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111**101*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*1101*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*1*0*10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*110*10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111***0110	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
111*1*0110	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
111**10110	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
111*110110	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
1111*1*010	0	0	0	0	0	0	0	0	0	0	0	0	-13	0	0
1101*0*0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111**100*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111***0010	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
111**10010	0	0	0	0	0	0	0	0	0	0	0	0	13	0	0
110**000*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
101*0*100	0	0	1	0	0	0	0	0	
10110*100	0	0	1	0	0	0	0	0	
000*0*1001	0	0	1	0	0	0	0	0	0
1101*0*00	0	0	1	0	0	0	0	0	
110**0100	0	0	1	0	0	0	0	0	
1101*0100	0	0	1	0	0	0	0	0	
000**01001	0	0	1	0	0	0	0	0	0
000*001001	0	0	1	0	0	0	0	0	0
000000*001	0	0	1	0	0	0	0	0	0
101*0*000	0	0	1	0	0	0	0	0	
110**0000	0	0	1	0	0	0	0	0	
100*00000	0	0	1	0	0	0	0	0	
111111*110	0	0	1	0	0	0	0	0	0
111*110**0	0	0	1	0	0	0	0	0	0
111**101*0	0	0	1	0	0	0	0	0	0
111*1101*0	0	0	1	0	0	0	0	0	0
111*1*0*10	0	0	1	0	0	0	0	0	0
111*110*10	0	0	1	0	0	0	0	0	0
111***0110	0	0	1	0	0	0	0	0	0
111*1*0110	0	0	1	0	0	0	0	0	0
111**10110	0	0	1	0	0	0	0	0	0
111*110110	0	0	1	0	0	0	0	0	0
1111*1*010	0	0	1	0	0	0	0	0	0
1101*0*0*0	0	0	1	0	0	0	0	0	0
111**100*0	0	0	1	0	0	0	0	0	0
111***0010	0	0	1	0	0	0	0	0	0
111**10010	0	0	1	0	0	0	0	0	0
110**000*0	0	0	1	0	0	0	0	0	0

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
001*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*11*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*11111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*1*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*11*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001111*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1*11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111*11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1*1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111*1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*11*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*1*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*11*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001111*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*1*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00111*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0011*11111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001*111111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*01111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*011*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01011*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*011111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*011**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01011**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*11*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*011*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01011*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*11111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
001*1	0	0	0	0					
0011*1	0	0	0	0	0				
001*11	0	0	0	0	0				
001*1*1	0	0	0	0	0	0			
00111*1	0	0	0	0	0	0			
0011*11	0	0	0	0	0	0			
001*111	0	0	0	0	0	0			
0011*1*1	0	0	0	0	0	0	0		
001*11*1	0	0	0	0	0	0	0		
001111*1	0	0	0	0	0	0	0		
001*1*11	0	0	0	0	0	0	0		
00111*11	0	0	0	0	0	0	0		
0011*111	0	0	0	0	0	0	0		
001*1111	0	0	0	0	0	0	0		
001*1*1*1	0	0	0	0	0	0	0	0	
00111*1*1	0	0	0	0	0	0	0	0	
0011*11*1	0	0	0	0	0	0	0	0	
001*111*1	0	0	0	0	0	0	0	0	
0011111*1	0	0	0	0	0	0	0	0	
0011*1*11	0	0	0	0	0	0	0	0	
001*11*11	0	0	0	0	0	0	0	0	
001111*11	0	0	0	0	0	0	0	0	
001*1*111	0	0	0	0	0	0	0	0	
00111*111	0	0	0	0	0	0	0	0	
0011*1111	0	0	0	0	0	0	0	0	
001*11111	0	0	0	0	0	0	0	0	
0011*1*1*1	0	0	0	0	0	0	0	0	0
001*11*1*1	0	0	0	0	0	0	0	0	0
001111*1*1	0	0	0	0	0	0	0	0	0
001*1*11*1	0	0	0	0	0	0	0	0	0
00111*11*1	0	0	0	0	0	0	0	0	0
0011*111*1	0	0	0	0	0	0	0	0	0
001*1111*1	0	0	0	0	0	0	0	0	0
00111111*1	0	0	0	0	0	0	0	0	0
001*1*1*11	0	0	0	0	0	0	0	0	0
00111*1*11	0	0	0	0	0	0	0	0	0
0011*11*11	0	0	0	0	0	0	0	0	0
001*111*11	0	0	0	0	0	0	0	0	0
0011111*11	0	0	0	0	0	0	0	0	0
0011*1*111	0	0	0	0	0	0	0	0	0
001*11*111	0	0	0	0	0	0	0	0	0
001111*111	0	0	0	0	0	0	0	0	0
001*1*1111	0	0	0	0	0	0	0	0	0
00111*1111	0	0	0	0	0	0	0	0	0
0011*11111	0	0	0	0	0	0	0	0	0
001*111111	0	0	0	0	0	0	0	0	0
0*011	0	0	0	0					
010*11	0	0	0	0	0				
0*0111	0	0	0	0	0				
010*111	0	0	0	0	0	0			
0*01111	0	0	0	0	0	0			
0*011*11	0	0	0	0	0	0	0		
01011*11	0	0	0	0	0	0	0		
010*1111	0	0	0	0	0	0	0		
0*011111	0	0	0	0	0	0	0		
0*011**11	0	0	0	0	0	0	0	0	
01011**11	0	0	0	0	0	0	0	0	
010*11*11	0	0	0	0	0	0	0	0	
0*0111*11	0	0	0	0	0	0	0	0	
010111*11	0	0	0	0	0	0	0	0	
0*011*111	0	0	0	0	0	0	0	0	
01011*111	0	0	0	0	0	0	0	0	
010*11111	0	0	0	0	0	0	0	0	

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0*0111111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*11**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0111**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010111**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*01111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0101111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*011**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01011**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*11*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0111*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010111*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*011*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01011*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*111111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*01111111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000111**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**1**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*1**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*11**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001111**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011**1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000111*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011111*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000111**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*11*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001111*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*1*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000111*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001**1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00011*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0001*11111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**0111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01*0111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
0*0111111	0	0	0	0	0	0	0	0	
010*11**11	0	0	0	0	0	0	0	0	0
0*0111**11	0	0	0	0	0	0	0	0	0
010111**11	0	0	0	0	0	0	0	0	0
010*111*11	0	0	0	0	0	0	0	0	0
0*01111*11	0	0	0	0	0	0	0	0	0
0101111*11	0	0	0	0	0	0	0	0	0
0*011**111	0	0	0	0	0	0	0	0	0
01011**111	0	0	0	0	0	0	0	0	0
010*11*111	0	0	0	0	0	0	0	0	0
0*0111*111	0	0	0	0	0	0	0	0	0
010111*111	0	0	0	0	0	0	0	0	0
0*011*1111	0	0	0	0	0	0	0	0	0
01011*1111	0	0	0	0	0	0	0	0	0
010*111111	0	0	0	0	0	0	0	0	0
0*01111111	0	0	0	0	0	0	0	0	0
0001*1	0	0	0	0	0				
0001**1	0	0	0	0	0	0			
00011*1	0	0	0	0	0	0			
0001*11	0	0	0	0	0	0			
00011**1	0	0	0	0	0	0	0		
0001*1*1	0	0	0	0	0	0	0		
000111*1	0	0	0	0	0	0	0		
0001**11	0	0	0	0	0	0	0		
00011*11	0	0	0	0	0	0	0		
0001*111	0	0	0	0	0	0	0		
0001*1**1	0	0	0	0	0	0	0	0	
000111**1	0	0	0	0	0	0	0	0	
0001**1*1	0	0	0	0	0	0	0	0	
00011*1*1	0	0	0	0	0	0	0	0	
0001*11*1	0	0	0	0	0	0	0	0	
0001111*1	0	0	0	0	0	0	0	0	
00011**11	0	0	0	0	0	0	0	0	
0001*1*11	0	0	0	0	0	0	0	0	
000111*11	0	0	0	0	0	0	0	0	
0001**111	0	0	0	0	0	0	0	0	
00011*111	0	0	0	0	0	0	0	0	
0001*1111	0	0	0	0	0	0	0	0	
0001**1**1	0	0	0	0	0	0	0	0	0
00011*1**1	0	0	0	0	0	0	0	0	0
0001*11**1	0	0	0	0	0	0	0	0	0
0001111**1	0	0	0	0	0	0	0	0	0
00011**1*1	0	0	0	0	0	0	0	0	0
0001*1*1*1	0	0	0	0	0	0	0	0	0
000111*1*1	0	0	0	0	0	0	0	0	0
0001**11*1	0	0	0	0	0	0	0	0	0
00011*11*1	0	0	0	0	0	0	0	0	0
0001*111*1	0	0	0	0	0	0	0	0	0
00011111*1	0	0	0	0	0	0	0	0	0
0001*1**11	0	0	0	0	0	0	0	0	0
000111**11	0	0	0	0	0	0	0	0	0
0001**1*11	0	0	0	0	0	0	0	0	0
00011*1*11	0	0	0	0	0	0	0	0	0
0001*11*11	0	0	0	0	0	0	0	0	0
0001111*11	0	0	0	0	0	0	0	0	0
00011**111	0	0	0	0	0	0	0	0	0
0001*1*111	0	0	0	0	0	0	0	0	0
000111*111	0	0	0	0	0	0	0	0	0
0001**1111	0	0	0	0	0	0	0	0	0
00011*1111	0	0	0	0	0	0	0	0	0
0001*11111	0	0	0	0	0	0	0	0	0
0**0111	0	0	0	0	0	0			
01*0111	0	0	0	0	0	0			

Minterms with One Cyclic Shift and whose related minterm contains '*'

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0*0*11**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*1**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*11**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*1**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*1*1**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*111**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*1**1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*11**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*1**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**0*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**0*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**0*11111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0000*1**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0000*1***1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0000*11**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0000*1**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0***0*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**0*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**0*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**00*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**00*1111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00000*1**1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0**0*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000**0*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**0*0*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**00*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**00*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0**000*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*0**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1**1*000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1111*0**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*00**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*0**00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11**1*000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1**11*000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1**1*0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1111*0***0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11111*0**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*0*0**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1111*00**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*000**0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11**11*0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*0**0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00*01*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00*01*1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**01*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*0**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0*01*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0*01**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*0**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0*01*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*01*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00*0*01*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**1*101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000**1*101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11*10*0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11**0*010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101*1**00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1*1*10*00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11**10*0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*10*0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111**0*010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00*01***11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
0*0*11**11	1	0	0	0	0	0	0	0	0
000*1**1	1	0	0	0	0	0	0		
000*11**1	1	0	0	0	0	0	0	0	
000*1**11	1	0	0	0	0	0	0	0	
000*1*1**1	1	0	0	0	0	0	0	0	0
000*111**1	1	0	0	0	0	0	0	0	0
000*1**1*1	1	0	0	0	0	0	0	0	0
000*11**11	1	0	0	0	0	0	0	0	0
000*1**111	1	0	0	0	0	0	0	0	0
0**0*111	1	0	0	0	0	0	0		
0**0*1111	1	0	0	0	0	0	0	0	
0**0*11111	1	0	0	0	0	0	0	0	0
0000*1**1	1	0	0	0	0	0	0	0	
0000*1***1	1	0	0	0	0	0	0	0	0
0000*11**1	1	0	0	0	0	0	0	0	0
0000*1**11	1	0	0	0	0	0	0	0	0
0***0*1111	1	0	0	0	0	0	0	0	0
00**0*111	1	0	0	0	0	0	0	0	
00**0*1111	1	0	0	0	0	0	0	0	0
0**00*111	1	0	0	0	0	0	0	0	
0**00*1111	1	0	0	0	0	0	0	0	0
00000*1**1	1	0	0	0	0	0	0	0	0
0*0**0*111	1	0	0	0	0	0	0	0	0
000**0*111	1	0	0	0	0	0	0	0	0
0**0*0*111	1	0	0	0	0	0	0	0	0
00**00*1*1	1	0	0	0	0	0	0	0	0
00**00*111	1	0	0	0	0	0	0	0	0
0**000*111	1	0	0	0	0	0	0	0	0
111*0**0	1	0	0	0	0	0	0		
1**1*000	1	0	0	0	0	0	0		
1111*0**0	1	0	0	0	0	0	0	0	
111*00**0	1	0	0	0	0	0	0	0	
111*0**00	1	0	0	0	0	0	0	0	
11**1*000	1	0	0	0	0	0	0	0	
1**11*000	1	0	0	0	0	0	0	0	
1**1*0000	1	0	0	0	0	0	0	0	
1111*0***0	1	0	0	0	0	0	0	0	0
11111*0**0	1	0	0	0	0	0	0	0	0
111*0*0**0	1	0	0	0	0	0	0	0	0
1111*00**0	1	0	0	0	0	0	0	0	0
111*000**0	1	0	0	0	0	0	0	0	0
11**11*0*0	1	0	0	0	0	0	0	0	0
111*0**0*0	1	0	0	0	0	0	0	0	0
00*01*1*1	0	1	0	0	0	0	0	0	
00*01*1*11	0	1	0	0	0	0	0	0	0
00**01*1*1	0	1	0	0	0	0	0	0	0
010*0**11	0	1	0	0	0	0	0	0	
0*0*01*11	0	1	0	0	0	0	0	0	
0*0*01**11	0	1	0	0	0	0	0	0	0
010*0**111	0	1	0	0	0	0	0	0	0
0*0*01*111	0	1	0	0	0	0	0	0	0
000*01*1*1	0	1	0	0	0	0	0	0	0
00*0*01*11	0	1	0	0	0	0	0	0	0
00**1*101	0	1	0	0	0	0	0	0	
000**1*101	0	1	0	0	0	0	0	0	0
11*10*0*0	0	1	0	0	0	0	0	0	
11**0*010	0	1	0	0	0	0	0	0	
101*1**00	0	1	0	0	0	0	0	0	
1*1*10*00	0	1	0	0	0	0	0	0	
11**10*0*0	0	1	0	0	0	0	0	0	0
111*10*0*0	0	1	0	0	0	0	0	0	0
111**0*010	0	1	0	0	0	0	0	0	0
00*01***11	0	1	0	0	0	0	0	0	0

Minterms with One Cyclic Shift

and whose related minterm contains '*'

Coefficients	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
010*0***11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*010***11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01010***11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00***01*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00***101*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00***1*101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00***10101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11***010*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11***0*010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11***01010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001**11*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001**11*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*00**011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00*00**011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110**00*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1*11**100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*10***111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01*01**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*10*1*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000***10*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*0*10*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000**01*01	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111***01*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111*1*01*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111**10*10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*1**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
010*1**111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
01*0***111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0010*1**11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0**011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00*0**011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0*0*0**011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000*0**011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1*1**100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11*1**100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
101*0**00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001**1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001**1*11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001**1*1*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
001**1*111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**0*101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
00**0*1011	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110**0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000***1*01	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
000**0*101	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11**1*010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110**0*00	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111***0*10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
111**1*010	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
110**0*0*0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Overlaps	1	2	3	4	5	6	7	8	9
010*0***11	0	1	0	0	0	0	0	0	0
0*010***11	0	1	0	0	0	0	0	0	0
01010***11	0	1	0	0	0	0	0	0	0
00***01*11	0	1	0	0	0	0	0	0	0
00***101*1	0	1	0	0	0	0	0	0	0
00***1*101	0	1	0	0	0	0	0	0	0
00***10101	0	1	0	0	0	0	0	0	0
11***010*0	0	1	0	0	0	0	0	0	0
11***0*010	0	1	0	0	0	0	0	0	0
11***01010	0	1	0	0	0	0	0	0	0
001**11*1	0	0	1	0	0	0	0	0	
001**11*11	0	0	1	0	0	0	0	0	0
0*00**011	0	0	1	0	0	0	0	0	
00*00**011	0	0	1	0	0	0	0	0	0
110**00*0	0	0	1	0	0	0	0	0	
1*11**100	0	0	1	0	0	0	0	0	
0*10***111	0	0	1	0	0	0	0	0	0
01*01**111	0	0	1	0	0	0	0	0	0
0*10*1*111	0	0	1	0	0	0	0	0	0
000***10*1	0	0	1	0	0	0	0	0	0
000*0*10*1	0	0	1	0	0	0	0	0	0
000**01*01	0	0	1	0	0	0	0	0	0
111***01*0	0	0	1	0	0	0	0	0	0
111*1*01*0	0	0	1	0	0	0	0	0	0
111**10*10	0	0	1	0	0	0	0	0	0
010*1**11	0	0	1	0	0	0	0	0	
010*1**111	0	0	1	0	0	0	0	0	0
01*0***111	0	0	1	0	0	0	0	0	0
0010*1**11	0	0	1	0	0	0	0	0	0
0*0**011	0	0	1	0	0	0	0		
00*0**011	0	0	1	0	0	0	0	0	
0*0*0**011	0	0	1	0	0	0	0	0	0
000*0**011	0	0	1	0	0	0	0	0	0
1*1**100	0	0	1	0	0	0	0		
11*1**100	0	0	1	0	0	0	0	0	
101*0**00	0	0	1	0	0	0	0	0	
001**1*1	0	0	1	0	0	0	0		
001**1*11	0	0	1	0	0	0	0	0	
001**1*1*1	0	0	1	0	0	0	0	0	0
001**1*111	0	0	1	0	0	0	0	0	0
00**0*101	0	0	1	0	0	0	0	0	
00**0*1011	0	0	1	0	0	0	0	0	0
110**0*0	0	0	1	0	0	0	0		
000***1*01	0	0	1	0	0	0	0	0	0
000**0*101	0	0	1	0	0	0	0	0	0
11**1*010	0	0	1	0	0	0	0	0	
110**0*00	0	0	1	0	0	0	0	0	
111***0*10	0	0	1	0	0	0	0	0	0
111**1*010	0	0	1	0	0	0	0	0	0
110**0*0*0	0	0	1	0	0	0	0	0	0

Bibliography

[1] Richard Beigel. The polynomial method in circuit complexity. In *Structure in Complexity Theory Conference, 1993., Proceedings of the Eighth Annual*, pages 82–95. IEEE, 1993.

[2] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288(1):21–43, 10/9 2002.

[3] Claude Carlet. Boolean functions for cryptography and error correcting codes. *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, 2:257, 2010.

[4] Sourav Chakraborty. On the sensitivity of cyclically-invariant Boolean functions. *Discrete Mathematics and Theoretical Computer Science*, 13(4):51–60, 2011.

[5] Parikshit Gopalan, Amir Shpilka, and Shachar Lovett. The complexity of Boolean functions in different characteristics. *Computational complexity*, 19(2):235–263, 2010.

[6] Jeff Kahn, M. Saks, and Dean Sturtevant. A topological approach to evasiveness. In *Foundations of Computer Science, 1983., 24th Annual Symposium on*, pages 31–33, 1983.

[7] Ryan O'Donnell. Some topics in analysis of Boolean functions. In *Proceedings of the 40th annual ACM symposium on Theory of computing*, pages 569–578. ACM, 2008.

[8] Ryan O'Donnell. *The Orthonormal Basis of Parity Functions*. Analysis of Boolean Functions. 2012.

[9] Abraham Silberschatz, Greg Gagne, and Peter B. Galvin. *Operating System Concepts*. J. Wiley & Sons, Hoboken, NJ, 7th edition, 2005.

[10] Edward L. Talmage and Amit Chakrabarti. On the Polynomial Degree of Minterm-Cyclic Functions. Senior Honors Thesis, Dartmouth College. 2012.