

A Matrix Approach for Constructing Quadratic APN Functions

Yuyin Yu, Mingsheng Wang, and Yongqiang Li

Abstract—We find a one to one correspondence between quadratic APN functions without linear and constant terms and a special kind of matrices (We call such matrices as QAMs). Based on the nice mathematical structures of the QAMs, we have developed efficient algorithms to construct quadratic APN functions. On \mathbb{F}_{2^7} , we have found more than 470 classes of new CCZ-inequivalent quadratic APN functions, which is 20 times more than the known ones. Before this paper, there are only 23 classes of CCZ-inequivalent APN functions on \mathbb{F}_{2^8} have been found. With our method, we have found more than 1000 classes of new CCZ-inequivalent quadratic APN functions, and this number is still arising quickly.

Index Terms—APN, quadratic functions, EA-equivalence, CCZ-equivalence.

I. INTRODUCTION

LOW differentially uniform permutations are very useful in cryptography, since they can provide good resistant against differential attack. For example, the Advanced Encryption Standard (AES) [11] uses a differentially 4 uniform permutation as its S-box (Substitution box). The differentially 4 uniform permutation is the best choice for now due to the lack of differentially 2 uniform permutations on \mathbb{F}_{2^8} . For clarity, we firstly introduce the definition of differential uniformity.

Definition 1: A mapping $F : \mathbb{F}_{2^n} \rightarrow \mathbb{F}_{2^n}$ is called differentially $\delta(F)$ uniform if

$$\delta(F) = \max_{a \in \mathbb{F}_{2^n}^*, b \in \mathbb{F}_{2^n}} \#\Delta_F(a, b),$$

where $\Delta_F(a, b) = \{x \in \mathbb{F}_{2^n} : F(x + a) + F(x) = b\}$ and $\#\Delta_F(a, b)$ is the cardinality of the set $\Delta_F(a, b)$. If $\delta(F) = 2$, we can say that F is **APN** (Almost perfect nonlinear).

Note that $\delta(F)$ is always even in \mathbb{F}_{2^n} , and APN functions provide optimal resistant against differential attack.

The terminology APN was introduced by Nyberg and Knudsen [20] in 1992. Carlet, Charpin and Zinoviev proved that if a function is APN, then its CCZ-equivalent functions are all APN. CCZ-equivalence is a generalization of EA-equivalence. In this paper, we only study quadratic functions. According to Yoshiara's results [21], two quadratic APN functions are CCZ-equivalent if and only if they are EA-equivalent.

Definition 2: Suppose F and F' are two functions from \mathbb{F}_{2^n} to \mathbb{F}_{2^n} , then

A short version of this paper has been submitted to WCC2013.

Y. Yu, M. Wang, and Y. Li are with State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (e-mail: yuyuin@163.com; mingsheng_wang@yahoo.com.cn; yongq.lee@gmail.com).

Y. Yu is also with University of Chinese Academy of Sciences, Beijing, China

1) F and F' are **EA-equivalent** (Extended affine equivalent) if

$$F'(x) = A_1(F(A_2(x))) + A_3(x),$$

where A_1 and A_2 are affine permutations on \mathbb{F}_{2^n} , and A_3 is an affine function on \mathbb{F}_{2^n} .

2) F and F' are **CCZ-equivalent** (Carlet-Charpin-Zinoviev equivalent) if there is an affine permutation which maps G_F to $G_{F'}$, where $G_F = \{(x, F(x)) : x \in \mathbb{F}_{2^n}\}$ is the graph of F , and $G_{F'}$ is the graph of F' .

An APN function is new if it is CCZ-inequivalent to any known ones. For a long time, finding new APN functions is an important topic in cryptography. In recent years, most of the new found APN functions are quadratic. Related work appeared in [1]-[8], [12]-[17], [22]. For a systematic knowledge of APN functions, the readers can turn to [9].

Our work was motivated by a recent breakthrough on APN functions. In 2009, Dillon et al. [3], [12], [4] found an APN permutation in dimension six, which is the first APN permutation in even dimension. Their idea can be summarized as checking whether there is a permutation CCZ-equivalent to the given APN function. Thus, if we want to find new APN permutations in even dimensions, we must find new APN functions at first.

Our aim is to find as many new APN functions as possible, especially on \mathbb{F}_{2^8} . Then, we will check whether these new APN functions are CCZ-equivalent to some permutations. If some function is CCZ-equivalent to a permutation, then there must exist an APN permutation.

Our main contributions can be summarized as follows. We find a one to one correspondence between restricted quadratic APN functions and QAMs. Thus, we only need to construct QAMs when we want to construct quadratic APN functions. These QAMs have nice mathematical structures. So, constructing QAMs is easier than constructing quadratic APN functions directly. We proved many properties of the QAMs in this paper. Based on these properties, we have designed some efficient algorithms to search the new APN functions. Before this paper, the scholars [12], [16] have found 19 and 23 classes of CCZ-inequivalent APN functions on \mathbb{F}_{2^7} and \mathbb{F}_{2^8} respectively. With our method, we have found more than 470 new CCZ-inequivalent APN functions on \mathbb{F}_{2^7} , and more than 1000 new CCZ-inequivalent quadratic APN functions on \mathbb{F}_{2^8} . The number of CCZ-inequivalent quadratic APN functions on \mathbb{F}_{2^8} is still arising quickly. We have checked all these new APN functions on \mathbb{F}_{2^8} , none of them is CCZ-equivalent to a permutation.

Our work is just a beginning, and we think that more useful

results will be obtained in the future.

II. NOTATIONS AND BASIC IDEAS

A. Notations

Let n be a positive integer, \mathbb{F}_{2^n} be the finite field with 2^n elements, and $\mathbb{F}_{2^n}[x]$ be the ring of polynomial in variable x .

Definition 3: Quadratic functions without linear and constant terms are called **restricted quadratic functions** in this paper.

Definition 4: [18] Two bases $\{\alpha_1, \dots, \alpha_n\}$ and $\{\theta_1, \theta_2, \dots, \theta_n\}$ of \mathbb{F}_{2^n} over \mathbb{F}_2 are said to be **dual bases** if for $1 \leq u, j \leq n$ we have

$$\text{Tr}(\alpha_u \theta_j) = \begin{cases} 0 & \text{for } u \neq j, \\ 1 & \text{for } u = j. \end{cases}$$

We will use the following convention and notations throughout the paper.

(i) For positive numbers r, s , $\mathbb{F}_{2^n}^{r \times s}$ denotes the space of $r \times s$ matrices over \mathbb{F}_{2^n} . For a matrix A , $A[i]$ denote the i th row of A , and $A[i, j]$ the (i, j) element of A ; moreover $B = \text{Submatrix}(A, 1, 1, r, c)$ denotes the $r \times c$ submatrix of A which consist of the first r rows and the first c columns.

(ii) Suppose $\{\alpha_1, \alpha_2, \dots, \alpha_n\}$ is a basis of \mathbb{F}_{2^n} over \mathbb{F}_2 , and $\{\theta_1, \theta_2, \dots, \theta_n\}$ is its dual basis. Let $M_\alpha \in \mathbb{F}_{2^n}^{n \times n}$ and $M_\theta \in \mathbb{F}_{2^n}^{n \times n}$ with $M_\alpha[i, u] = \alpha_u^{2^{i-1}}$ and $M_\theta[i, u] = \theta_u^{2^{i-1}}$ for $1 \leq u, i \leq n$. Then $M_\alpha^t M_\theta = (\text{Tr}(\alpha_u \theta_j))_{n \times n}$ for $1 \leq u, j \leq n$, so $M_\alpha^t M_\theta = I_n$, where I_n is the identity matrix of order n . Thus $M_\theta^{-1} = M_\alpha^t$ (M_α^t is the transpose of M_α).

(iii) Let $\eta_1, \eta_2, \dots, \eta_m$ be m elements on \mathbb{F}_{2^n} ($m, n \geq 1$), and $B = (\eta_1, \eta_2, \dots, \eta_m) \in \mathbb{F}_{2^n}^m$. $\text{Span}(B) = \text{Span}(\eta_1, \eta_2, \dots, \eta_m)$ denotes the subspace spanned by $\{\eta_1, \eta_2, \dots, \eta_m\}$ over \mathbb{F}_2 . $\text{Rank}_{\mathbb{F}_2}(B) = \text{Rank}_{\mathbb{F}_2}\{\eta_1, \eta_2, \dots, \eta_m\}$ is the dimension of $\text{Span}(B)$ over \mathbb{F}_2 , which we call the rank of B (over \mathbb{F}_2). Suppose $\eta_i = \sum_{j=1}^n \lambda_{i,j} \alpha_j$ for $1 \leq i \leq m$, where $\lambda_{i,j} \in \mathbb{F}_2$ for all i, j . Define a $m \times n$ matrix $\Lambda = (\lambda_{i,j})_{m \times n}$. Then $\text{Rank}_{\mathbb{F}_2}\{\eta_1, \eta_2, \dots, \eta_m\}$ equals to the rank of Λ .

Definition 5: Let H be an $n \times n$ matrix on \mathbb{F}_{2^n} . Then, the matrix H is called a **QAM** (quadratic APN matrix) if

- 1) H is a symmetric matrix with main diagonal elements all zeros;
- 2) Every nonzero linear combination of the n rows (or “columns” since H is symmetric) of H has rank $n - 1$.

B. One to One Correspondence between Restricted Quadratic APN Functions and QAMs

Let $F(x) = \sum_{1 \leq t < i \leq n} c_{i,t} x^{2^{i-1} + 2^{t-1}} \in \mathbb{F}_{2^n}[x]$ be a restricted quadratic function, and C_F be an $n \times n$ matrix with $C_F[t, i] = C_F[i, t] = c_{i,t}$ for $1 \leq t < i \leq n$ and $C_F[i, i] = 0$ for $1 \leq i \leq n$.

Given a basis $\alpha = \{\alpha_1, \dots, \alpha_n\}$ for \mathbb{F}_{2^n} over \mathbb{F}_2 , and let $M = M_\alpha$, $\Theta = M_\theta$. For any restricted quadratic function $F(x)$, let $H = M^t C_F M$. Then H is a symmetric matrix over \mathbb{F}_{2^n} with main diagonal elements zeros.

Conversely, for a symmetric matrix H with main diagonal elements all zeros, we can define a unique restricted quadratic

function $F(x)$ such that $H = M^t C_F M$. $F(x)$ is called the quadratic function defined by H relative to the ordered basis α .

In order to prove Theorem 1, let's show how to denote restricted quadratic functions in matrix form.

Let $F(x) = \sum_{1 \leq t < i \leq n} c_{i,t} x^{2^{i-1} + 2^{t-1}} \in \mathbb{F}_{2^n}[x]$, and an $n \times n$ matrix $E = (e_{i,t})_{n \times n}$ be defined by

$$e_{i,t} = \begin{cases} c_{i,t} & \text{if } i > t, \\ 0 & \text{if } i \leq t. \end{cases} \quad (1)$$

Let $X = (x^{2^0}, x^{2^1}, \dots, x^{2^{n-1}})^t$. Then we have

$$F(x) = X^t E X. \quad (2)$$

Let $x = x_1 \alpha_1 + x_2 \alpha_2 + \dots + x_n \alpha_n$, where $x_i \in \mathbb{F}_2$, $1 \leq i \leq n$. Then (2) equals to

$$F(x) = \bar{x}^t M^t E M \bar{x}, \quad (3)$$

where $\bar{x} = (x_1, x_2, \dots, x_n)^t$, and $M = M_\alpha$.

Based on (3), we can build a one to one correspondence between restricted quadratic APN functions and QAMs.

Theorem 1: Let $F(x) = \sum_{1 \leq t < i \leq n} c_{i,t} x^{2^{i-1} + 2^{t-1}} \in \mathbb{F}_{2^n}[x]$, C_F and M be defined as above, and $H = M^t C_F M$. Then, $\delta(F) = 2^k$ if and only if any nonzero linear combination of the n rows of H has rank at least $n - k$. In particular, F is APN on \mathbb{F}_{2^n} if and only if H is a QAM.

Proof: Let E and \bar{x} be the same as in (1) and (3), and $a = a_1 \alpha_1 + a_2 \alpha_2 + \dots + a_n \alpha_n$, where $\bar{a} = (a_1, \dots, a_n) \in \mathbb{F}_2^n \setminus \{0\}$. Let

$$D_a(x) = F(x + a) + F(x) + F(a).$$

Note that $D_a(x)$ is a linear function. So $\delta(F) = 2^k$ if and only if $\max\{\dim_{\mathbb{F}_2}(\text{Ker}(D_a)) \mid a \in \mathbb{F}_{2^n}^*\} = k$. Based on (3), we have

$$\begin{aligned} D_a(x) &= (\bar{x} + \bar{a})^t M^t E M (\bar{x} + \bar{a}) + \bar{x}^t M^t E M \bar{x} \\ &\quad + \bar{a}^t M^t E M \bar{a} \\ &= \bar{x}^t M^t E M (\bar{x} + \bar{a} + \bar{x}) + \bar{a}^t M^t E M (\bar{x} + \bar{a} + \bar{x}) \\ &= \bar{x}^t M^t E M \bar{a} + (\bar{a}^t M^t E M \bar{x})^t \\ &= \bar{x}^t M^t (E + E^t) M \bar{a} \\ &= \bar{x}^t M^t C_F M \bar{a} \\ &= \bar{x}^t H \bar{a}. \end{aligned}$$

Hence $\dim_{\mathbb{F}_2}(\text{Ker}(D_a)) = k$ if and only if $\text{Rank}_{\mathbb{F}_2}((H \bar{a})^t) = n - k$. $H \bar{a}$ is a nonzero linear combination of the n columns of H since $\bar{a} \in \mathbb{F}_2^n \setminus \{0\}$. Thus, $\delta(F) = 2^k$ if and only if any nonzero linear combination of the n columns of H has rank at least $n - k$.

Note that H is symmetric, thus the above results implies this theorem. ■

The matrix H which is associated with $F(x)$ in Theorem 1 is called the matrix of $F(x)$ relative to the ordered bases $\{\alpha_1, \dots, \alpha_n\}$.

Note that M is an invertible matrix on \mathbb{F}_{2^n} , so the correspondence between restricted quadratic APN functions and

QAMs is one to one. It seems that another similar approach have been considered by Knuth and Edel, the readers can turn to Edel's ppt [13] for details.

Theorem 1 is very useful when we want to study the differential properties of quadratic functions, and it can be generalized to \mathbb{F}_{p^n} , where p is any prime and n is a positive integer. In this paper, we use only this theorem to study quadratic APN functions on \mathbb{F}_{2^n} .

III. PROPERTIES OF QAMs

We will introduce some properties of QAMs in this section. Let $F(x)$ be a given quadratic function. First we should like to inquire what happens to corresponding matrices when the ordered basis is changed. Let $\alpha = \{\alpha_1, \dots, \alpha_n\}$ and $\beta = \{\beta_1, \dots, \beta_n\}$ be two ordered bases for \mathbb{F}_{2^n} over \mathbb{F}_2 . Assume H_α and H_β are corresponding matrices for $F(x)$ relative to the α, β , respectively. How are the matrices H_α and H_β related?

As we know, there is a unique invertible $n \times n$ matrix $P \in \mathbb{F}_2^{n \times n}$ such that

$$(\beta_1, \dots, \beta_n) = (\alpha_1, \dots, \alpha_n)P.$$

Hence we have $M_\beta = M_\alpha P$. So we have $H_\beta = M_\beta^t C_F M_\beta = P^t H_\alpha P$.

Conversely, assume that H', H are two $n \times n$ symmetric matrices with main diagonal elements all zeros such that $H' = P^t H P$ for an invertible matrix P over \mathbb{F}_2 . Let $F(x)$ be the quadratic function defined by H relative to the ordered basis α . Let $\gamma = \{\gamma_1, \dots, \gamma_n\}$ be defined by $(\gamma_1, \dots, \gamma_n) = (\alpha_1, \dots, \alpha_n)P$. Then γ be a basis for \mathbb{F}_{2^n} , and $F(x)$ is also the quadratic function defined by H' relative to ordered basis γ . Now let $F'(x)$ be the quadratic function defined by H' relative to α , then how are the functions $F(x)$ and $F'(x)$ related?

In order to answer this problem, we first give a lemma in the following.

Lemma 1: Suppose $H = (h_{u,v})_{n \times n}$ is a symmetric matrix over \mathbb{F}_{2^n} with $h_{u,u} = 0$ for all $1 \leq u \leq n$. Define a set $S = \{K = (k_{u,v})_{n \times n} \mid k_{u,v} + k_{v,u} = h_{u,v} \text{ for all } 1 \leq v \leq u \leq n\}$. Then

- 1) $T \in S$ if and only if $T + T^t = H$.
- 2) If $T_1 + T_1^t = H$ and $T_2 + T_2^t = H$, then there exists a symmetric matrix A such that $T_2 = T_1 + A$.

Proof: 1) is obvious, omitting it, we prove only 2) in the following. Let $T_1 + T_1^t = H$ and $T_2 + T_2^t = H$. Then for any symmetry matrix A , there is

$$(T_1 + A) + (T_1 + A)^t = T_1 + T_1^t + A + A^t = T_1 + T_1^t = H,$$

which implies that $T_1 + A \in S$ for any symmetric matrix A .

Define a set $S' = \{T_1 + A \mid A \text{ is symmetry}\}$. Then $\#S' = 2^{n^2(n+1)/2} = \#S$, and for any $T \in S'$, we have $T + T^t = H$. Thus there must be $S = S'$. Therefore, $T_2 \in S'$, which implies that there exists a symmetric matrix A such that $T_2 = T_1 + A$. ■

Theorem 2: Let $H \in \mathbb{F}_{2^n}^{n \times n}$ be a symmetric matrix with main diagonal elements all zeros, and $P \in \mathbb{F}_2^{n \times n}$ be an invertible matrix. Suppose $H' = P^t H P$, then the quadratic functions defined by H and H' relative to an ordered basis α are EA-equivalent. Especially, H is a QAM if and only if H' is a QAM.

Proof: Let $F(x)$ and $F'(x)$ be the functions defined by H and H' relative to α , respectively, where $F(x) = \sum_{1 \leq t < i \leq n} c_{i,t} x^{2^{i-1}+2^{t-1}}$, and $F'(x) = \sum_{1 \leq t < i \leq n} c'_{i,t} x^{2^{i-1}+2^{t-1}}$. Let $E = (e_{i,t})$, and $E' = (e'_{i,t})$ be two $n \times n$ matrices with

$$e_{i,t} = \begin{cases} c_{i,t} & \text{if } i > t \\ 0 & \text{if } i \leq t, \end{cases} \quad \text{and } e'_{i,t} = \begin{cases} c'_{i,t} & \text{if } i > t \\ 0 & \text{if } i \leq t. \end{cases}$$

By (3), we have

$$F(x) = \bar{x}^t M^t E M \bar{x}, \text{ and } F'(x) = \bar{x}^t M^t E' M \bar{x},$$

where $\bar{x} = (x_1, x_2, \dots, x_n)^t \in \mathbb{F}_2^n$.

Let $T = M^t E M$, and $T' = M^t E' M$. Then $T + T^t = H$, and $T' + T'^t = H'$, which implies that $P^t T P + P^t T^t P = P^t H P = H' = T' + T'^t$. According to Lemma 1, there exists a symmetric matrix $A = (a_{u,v})_{n \times n}$ such that $T' = P^t T P + A$. Thus we have $\bar{x}^t T' \bar{x} = \bar{x}^t (P^t T P + A) \bar{x}$. Hence

$$\begin{aligned} F'(x) &= \bar{x}^t M^t E' M \bar{x} = \bar{x}^t P^t M^t E M P \bar{x} + \bar{x}^t A \bar{x} \\ &= G(x) + \bar{x}^t A \bar{x}, \end{aligned} \quad (4)$$

where $G(x) = \bar{x}^t P^t M^t E M P \bar{x}$.

Since A is symmetric, we have

$$\bar{x}^t A \bar{x} = \sum_{u=1}^n \sum_{v=1}^n a_{u,v} x_u x_v = \sum_{u=1}^n a_{u,u} x_u^2 = \sum_{u=1}^n a_{u,u} x_u. \quad (5)$$

By (4) and (5), $F'(x)$ is EA-equivalent to $G(x)$. As for $G(x)$, we have

$$G(x) = \bar{x}^t P^t M^t E M P \bar{x} = \bar{y}^t M^t E M \bar{y} = F(y),$$

where $\bar{y} = (y_1, y_2, \dots, y_n)^t = P \bar{x}$. So $G(x)$ is EA-equivalent to $F(x)$. Thus it can be deduced that $F'(x)$ is EA-equivalent to $F(x)$. Thus $\delta(F') = \delta(F)$, and then the whole theorem is proved. ■

We need the following lemma (which is Theorem 2.3 in [19]) when proving Lemma 3.

Lemma 2: [19] Let $\{\theta_1, \theta_2, \dots, \theta_n\}$ be any given basis of \mathbb{F}_{2^n} over \mathbb{F}_2 , and let $L(x)$ be linearized polynomial over \mathbb{F}_{2^n} . Then there exists a unique vector $(\beta_1, \beta_2, \dots, \beta_n) \in \mathbb{F}_{2^n}^n$ such that

$$L(x) = \sum_{j=1}^n \text{Tr}(\theta_j x) \beta_j = \sum_{i=1}^n \left(\sum_{j=1}^n \beta_j \theta_j^{2^{i-1}} \right) x^{2^{i-1}}.$$

Moreover, let k be an integer such that $0 \leq k \leq n$, then $\dim_{\mathbb{F}_2}(\text{Ker}(L)) = k$ if and only if $\text{Rank}_{\mathbb{F}_2}\{\beta_1, \beta_2, \dots, \beta_n\} = n - k$.

Lemma 3: Every quadratic function $Q(x) \in \mathbb{F}_{2^n}[x]$ with $Q(0) = 0$ can be denoted as

$$\begin{aligned} Q(x) &= \sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) (\eta_{u,v} + \eta_{v,u}) \\ &\quad + \sum_{u=1}^n \text{Tr}(\theta_u x) \eta_{u,u} \end{aligned} \quad (6)$$

$$= \sum_{1 \leq t < i \leq n} c_{i,t} x^{2^{i-1}+2^{t-1}} + \text{Lin}(x), \quad (7)$$

where

$$c_{i,t} = \sum_{1 \leq u,v \leq n} \theta_u^{2^{i-1}} \theta_v^{2^{t-1}} (\eta_{v,u} + \eta_{u,v}),$$

$$\begin{aligned} \text{Lin}(x) &= \sum_{1 \leq v < u \leq n} (\eta_{u,v} + \eta_{v,u}) \text{Tr}(\theta_u \theta_v x^2) \\ &\quad + \sum_{u=1}^n \text{Tr}(\theta_u x) \eta_{u,u}, \end{aligned}$$

and

$$\eta_{u,v} \ (1 \leq u, v \leq n) \in \mathbb{F}_{2^n}.$$

Proof: According to Lemma 2, every quadratic function without constant term can be denoted as $Q(x) = \sum_{t=1}^n L'_t(x) x^{2^{t-1}}$, where $L'_t(x) = \sum_{u=1}^n \text{Tr}(\theta_u x) \omega_{u,t}$ and $\omega_{u,t} \in \mathbb{F}_{2^n}$ for all $1 \leq u, t \leq n$. Then we have $Q(x) = \sum_{u=1}^n L_u(x) \text{Tr}(\theta_u x)$, where $L_u(x) = \sum_{t=1}^n \omega_{u,t} x^{2^{t-1}}$. Again, according to Lemma 2, we have $L_u(x) = \sum_{v=1}^n \text{Tr}(\theta_v x) \eta_{v,u}$, where $\eta_{v,u} \in \mathbb{F}_{2^n}$ such that $\omega_{u,t} = \sum_{v=1}^n \theta_v^{2^{t-1}} \eta_{v,u}$. Hence we have

$$\begin{aligned} Q(x) &= \sum_{u=1}^n L_u(x) \text{Tr}(\theta_u x) \\ &= \sum_{u=1}^n \left(\sum_{v=1}^n \text{Tr}(\theta_v x) \eta_{v,u} \right) \text{Tr}(\theta_u x) \\ &= \sum_{1 \leq u, v \leq n} \text{Tr}(\theta_u x) \eta_{u,v} \text{Tr}(\theta_v x) \\ &= \sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) (\eta_{u,v} + \eta_{v,u}) \\ &\quad + \sum_{u=1}^n \text{Tr}(\theta_u x) \eta_{u,u} \\ &= \sum_{1 \leq v < u \leq n} \sum_{i=1}^n (\theta_u x)^{2^{i-1}} \sum_{t=1}^n (\theta_u x)^{2^{t-1}} (\eta_{u,v} + \eta_{v,u}) \\ &\quad + \sum_{u=1}^n \text{Tr}(\theta_u x) \eta_{u,u} \\ &= \sum_{1 \leq v < u \leq n} \sum_{1 \leq i, t \leq n} ((\eta_{u,v} \\ &\quad + \eta_{v,u}) \theta_u^{2^{i-1}} \theta_v^{2^{t-1}} x^{2^{i-1}+2^{t-1}}) + \sum_{u=1}^n \text{Tr}(\theta_u x) \eta_{u,u} \\ &= \sum_{1 \leq t < i \leq n} \sum_{1 \leq v < u \leq n} ((\eta_{u,v} + \eta_{v,u}) (\theta_u^{2^{i-1}} \theta_v^{2^{t-1}} \\ &\quad + \theta_u^{2^{t-1}} \theta_v^{2^{i-1}}) x^{2^{i-1}+2^{t-1}}) + \text{Lin}(x) \\ &= \sum_{1 \leq t < i \leq n} \left(\sum_{1 \leq u, v \leq n} \theta_u^{2^{i-1}} \theta_v^{2^{t-1}} (\eta_{v,u} \right. \\ &\quad \left. + \eta_{u,v}) \right) x^{2^{i-1}+2^{t-1}} + \text{Lin}(x) \\ &= \sum_{1 \leq t < i \leq n} c_{i,t} x^{2^{i-1}+2^{t-1}} + \text{Lin}(x). \end{aligned}$$

Lemma 3 will help us to prove the following result.

Theorem 3: Let $H = (h_{u,v}) \in \mathbb{F}_{2^n}^{n \times n}$ be a symmetric matrix with main diagonal elements all zeros, and L be a linear

permutation on \mathbb{F}_{2^n} . Let $H' = (h'_{u,v}) \in \mathbb{F}_{2^n}^{n \times n}$ such that $h'_{u,v} = L(h_{u,v})$ for all $1 \leq u, v \leq n$. Then the quadratic functions defined by H and H' relative to α are EA-equivalent. In particular, H is a QAM if and only if H' is a QAM.

Proof: Suppose the corresponding functions of H and H' are $F(x) = \sum_{1 \leq t < i \leq n} c_{i,t} x^{2^{i-1}+2^{t-1}}$ and $F'(x) = \sum_{1 \leq t < i \leq n} c'_{i,t} x^{2^{i-1}+2^{t-1}}$ respectively.

Let C_F be the same as in Section II. Then $H = M^t C_F M$. Hence $C_F = (M^t)^{-1} H M^{-1} = \Theta H \Theta^t$ ($\Theta = M_\theta$, see Section II). So

$$c_{i,t} = \sum_{1 \leq u, v \leq n} \theta_u^{2^{i-1}} \theta_v^{2^{t-1}} h_{u,v}.$$

Choose $\eta_{u,v}$ such that $\eta_{u,v} + \eta_{v,u} = h_{u,v}$ for all $1 \leq u, v \leq n$, and let $\eta_{u,u} = 0$ for all $1 \leq u \leq n$. Define a quadratic function $Q(x)$ over \mathbb{F}_2^n as follows:

$$\begin{aligned} Q(x) &= \sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) h_{u,v} \\ &= \sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) (\eta_{u,v} + \eta_{v,u}). \end{aligned}$$

Then from the proof of Lemma 3, we have

$$Q(x) = F(x) + \text{Lin}(x), \quad (8)$$

for some linear function $\text{Lin}(x)$ over \mathbb{F}_{2^n} .

Furthermore we define $Q'(x)$ by

$$Q'(x) = \sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) h'_{u,v}. \quad (9)$$

Using the same reasoning as $Q(x)$ and $F(x)$, we get $Q'(x) = F'(x) + \text{Lin}'(x)$ for some linear function $\text{Lin}'(x)$ over \mathbb{F}_{2^n} .

Thus we have

$$\begin{aligned} Q'(x) &= \sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) h'_{u,v} \\ &= \sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) L(h_{u,v}) \\ &= L \left(\sum_{1 \leq v < u \leq n} \text{Tr}(\theta_u x) \text{Tr}(\theta_v x) h_{u,v} \right) \\ &= L(Q(x)). \end{aligned} \quad (10)$$

By (8), (9) and (10), it deduce that $F(x)$ and $F'(x)$ are EA-equivalent. ■

Remark 1: Let $H = (h_{u,v})_{n \times n}$ and $H' = (h'_{u,v})_{n \times n}$ be two $n \times n$ matrices, and L be linear permutation on \mathbb{F}_{2^n} , then $H' = L(H)$ means $h'_{u,v} = L(h_{u,v})$ for all $1 \leq u, v \leq n$.

Based on Theorem 2 and Theorem 3 we can obtain the following corollary.

Corollary 1: Let $F(x)$ and $F'(x)$ be two restricted quadratic functions, and H and H' be their corresponding matrices, respectively. Then $F(x)$ is EA-equivalent to $F'(x)$ if $H' = L(P^t H P)$, where P is some $n \times n$ invertible matrix over \mathbb{F}_2 , and L is a linear permutation over \mathbb{F}_{2^n} .

Up to now, we have introduced the main theoretical results. We can greatly reduce the data complexity when we want to find all the CCZ-inequivalent quadratic APN functions. In the following section, we will describe how to find great amount of CCZ-inequivalent quadratic APN functions.

IV. CONSTRUCTING QUADRATIC APN FUNCTIONS FROM A GIVEN QAM

In this section, we will introduce how to construct QAMs (or, restricted quadratic APN functions).

A. Further properties of QAMs

In Section III, we have introduced some theoretical results. In this section, we will give more results on QAMs. These results will be useful for designing effective algorithms for constructing quadratic APN functions.

Lemma 4: Let $H \in \mathbb{F}_{2^n}^{n \times n}$ be a symmetric matrix with main diagonal elements all zeros. Then every nonzero linear combination of the n rows of H has rank at most $n - 1$.

Proof: Obviously, $H[i]$ has rank at most $n - 1$ for any $1 \leq i \leq n$. Suppose $\mu = H[i_1] + H[i_2] + \dots + H[i_t]$, where $2 \leq t \leq n$ and $\{i_1, i_2, \dots, i_t\}$ is a subset of $\{1, 2, \dots, n\}$. Then we have $\mu[i_1] + \mu[i_2] + \dots + \mu[i_t] = 0$, so $\text{Rank}_{\mathbb{F}_2}(\mu) \leq n - 1$, which implies this lemma. ■

Before proving the main result of this section, let us give the following definition which is convenient for our discussion.

Definition 6: Let $H \in \mathbb{F}_{2^n}^{m \times k}$ ($m, k \leq n$). H is called proper if every nonzero linear combination of the m rows of H has rank at least $k - 1$.

First, we give the following lemma.

Lemma 5: Let $A \in \mathbb{F}_{2^n}^{r \times c}$ ($1 \leq r < c \leq n$), and $A' = AP$, where $P \in \mathbb{F}_2^{c \times c}$ is invertible. Then A is proper implies that A' is also proper.

Proof: Let $S = \{\sum_{i=1}^r \lambda_i A[i] : (\lambda_1, \dots, \lambda_r) \in \mathbb{F}_2^r \setminus \{0\}\}$, and $S' = \{\sum_{i=1}^r \lambda_i A'[i] : (\lambda_1, \dots, \lambda_r) \in \mathbb{F}_2^r \setminus \{0\}\}$. Let $S'' = \{sP : s \in S\}$. Since P is invertible, we have $S' = S''$, and $\text{Rank}_{\mathbb{F}_2}(s) = \text{Rank}_{\mathbb{F}_2}(sP)$. ■

Now we can prove the following theorem:

Theorem 4: Let $A = (a_{i,j}) \in \mathbb{F}_{2^n}^{r \times c}$ ($1 \leq r < c \leq n$) such that $a_{i,j} = a_{j,i}$ and $a_{i,i} = 0$ for $1 \leq i, j \leq r$. Let $A[\cdot, k]$ be the k -th column of A , and $b = \sum_{k=1}^c \lambda_k A[\cdot, k]$, where $\lambda_k \in \mathbb{F}_2$ for $1 \leq k \leq c$. Assume $t = \text{Rank}_{\mathbb{F}_2}\{b[1], b[2], \dots, b[r]\}$. If A is proper, then we have:

- i) If $(\lambda_{r+1}, \dots, \lambda_c) = 0$, then $t = r - 1$;
- ii) If $(\lambda_{r+1}, \dots, \lambda_c) \neq 0$, then $t = r$.

Proof: i) Assume $(\lambda_{r+1}, \dots, \lambda_c) = 0$. Then $b = \sum_{k=1}^r \lambda_k A[\cdot, k]$. By Lemma 4, $t \leq r - 1$. Let $B = \text{Submatrix}(A, 1, 1, r, r)$, by the definition of A , we have

$$\begin{aligned} \text{Rank}_{\mathbb{F}_2}(\sum_{k=1}^r \lambda_k A[\cdot, k]) &= \text{Rank}_{\mathbb{F}_2}(\sum_{k=1}^r \lambda_k B[\cdot, k]) \\ &= \text{Rank}_{\mathbb{F}_2}(\sum_{k=1}^r \lambda_k B[k]). \end{aligned}$$

If $t < r - 1$, then we have $\text{Rank}_{\mathbb{F}_2}(\sum_{k=1}^r \lambda_k A[k]) < r - 1 + (c - r) = c - 1$, which contradicts with A being proper. Thus $t = r - 1$.

ii) Suppose $(\lambda_{r+1}, \dots, \lambda_c) \neq 0$. Without loss of generality, let $\lambda_c = 1$. Then substitute $A[\cdot, c]$ with b , we get a new $r \times c$ matrix A' . If $t < r$, then there exists $0 \neq (\lambda'_1, \dots, \lambda'_r) \in \mathbb{F}_2^r$ such that $\lambda'_1 A'[1, c] + \lambda'_2 A'[2, c] + \dots + \lambda'_r A'[r, c] = 0$. Without loss of generality, suppose $\lambda'_1 \neq 0$. Next we perform the following operations step by step, first, substitute $A'[1]$

with $\sum_{i=1}^r \lambda'_i A'[i]$ and get a new matrix A'' ; second, substitute $A''[\cdot, 1]$ with $\sum_{i=1}^r \lambda'_i A'[\cdot, i]$ and get a new matrix A''' . By Lemma 5 and the definition of proper, it implies that A' , A'' and A''' are also proper. However, after these changes, we have $A'''[1, 1] = A'''[1, c] = 0$, which contradicts with A''' being proper. Thus $t < r$ is not true, so $t = r$. ■

According to Theorem 4, we get the following corollary.

Corollary 2: Let $H = (h_{u,v})_{n \times n}$ be an $n \times n$ symmetric matrix over \mathbb{F}_{2^n} , and $A = \text{Submatrix}(H, 1, 1, r, c)$. Suppose $B = A^t = \text{Submatrix}(H, 1, 1, c, r)$. Then A is proper implies that B is also proper.

Corollary 2 is useful in our algorithm for constructing QAMs. Our algorithm can be summarized as “Guess and determine”. The basic idea is: every submatrix of a QAM must be proper (See Definition 6). So, if a matrix has a submatrix which is not proper, it cannot be a QAM. Based on this corollary, we can exclude some improper candidates in advance when we haven't known the whole values of the matrix. Notice that every QAM is symmetric, so we will know the values of $A = \text{Submatrix}(H, 1, 1, r, c)$ and $B = A^t = \text{Submatrix}(H, 1, 1, c, r)$ at the same time. According to Corollary 2, we only need to check whether A is proper. Thus we can avoid some unnecessary checking in our searching algorithm.

B. How to construct QAMs

In this section, we will introduce a problem and then we will show how to construct QAMs through solving this problem.

Problem 1: Suppose e_i is a vector of length n with $e_i[i] = 1$ and $e_i[j] = 0$ for $i \neq j$. Let x_i be a variable in \mathbb{F}_{2^n} , $i = 1, \dots, n - 1$. Suppose these $n - 1$ variables satisfy

$$\lambda_1 x_1 + \dots + \lambda_{n-1} x_{n-1} \in S_{\lambda_1 e_1 + \dots + \lambda_{n-1} e_{n-1}}, \quad (11)$$

for all $(\lambda_1, \dots, \lambda_{n-1}) \in \mathbb{F}_2^{n-1} \setminus \{0\}$, where $S_{\lambda_1 e_1 + \dots + \lambda_{n-1} e_{n-1}}$ are some subsets of \mathbb{F}_{2^n} . The problem is, how to find all $\vec{x} = (x_1, \dots, x_{n-1}) \in \mathbb{F}_2^{n-1}$ which satisfies (11).

(11) consists of $2^{n-1} - 1$ conditions, and we need to find all the qualified \vec{x} . As a matter of fact, all the constructions of QAMs can be reduced as Problem 1. Details are in the following.

Given an $n \times n$ QAM matrix H over \mathbb{F}_{2^n} , we wish to reassign the values of the last column of H to get some new QAMs. Let $A = \text{Submatrix}(H, 1, 1, n - 1, n - 1)$, it is easy to see that A is proper. By Lemma 4, any nonzero linear combination of the $n - 1$ rows of A has rank $n - 2$.

Let $c = (x_1, \dots, x_{n-1})^t$, and $H' = \begin{pmatrix} A & c \\ c^t & 0 \end{pmatrix}$. We want to choose suitable c to make H' a QAM. Actually, by Theorem 4 (ii), we need only to choose $c = (x_1, \dots, x_{n-1})^t$ to satisfy (11), where $S_{\lambda_1 e_1 + \dots + \lambda_{n-1} e_{n-1}} = \mathbb{F}_{2^n} \setminus \text{Span}(\lambda_1 A[1] + \dots + \lambda_{n-1} A[n - 1])$.

We can shrink S_{e_1} in (11). Let $V = \text{Span}(A[1, 1], A[1, 2], \dots, A[1, n - 1])$. In (11), $S_{e_1} = \mathbb{F}_{2^n} \setminus V$, which equals to $(V + a_1) \cup (V + a_2) \cup (V + a_3)$ for some $a_i \in \mathbb{F}_{2^n}$, $1 \leq i \leq 3$ because of $\dim(V) = n - 2$. Since $x_1 \in S_{e_1}$, there exists $y \in V$ such that $x_1 = y + a_i$ for some i , i.e., $a_i = x_1 + y$. Since $y \in V$ and $A[1, 1] = 0$, $y = \lambda_2 A[1, 2] + \dots + \lambda_{n-1} A[1, n - 1]$ for

some $\lambda_i \in \mathbb{F}_2$, $i = 2, \dots, n - 1$. So we may perform suitable column transformations to change x_1 into a_i , and perform the corresponding row transformations to change $H'[n, 1]$ into a_i . Since we consider only to find CCZ-inequivalent functions, then by Theorem 2, we may take $S_{e_1} = \{a_1, a_2, a_3\}$. Because in the above transformation, we do not use the first column, based on the same reason as the S_{e_1} , we may take $S_{e_2} = \{b_1, \dots, b_l\}$, where $l = 2^{n-1} - 2^{n-3}$.

Further, given a QAM H , we may also reassign the values of the last two columns of H to get some new QAMs. This can also be reduced to Problem 1. The difference is that we must apply the problem twice. Similarly, we can reassign more columns of H . So this method can generate almost all CCZ-inequivalent quadratic APN functions if we change enough columns.

In view of the above discussions, an algorithm for solving problem 1 is important for our approach for constructing new quadratic functions. In the following, we describe an algorithm for solving Problem 1.

Algorithm 1: Step 1. Initialization. Given a QAM H over \mathbb{F}_{2^n} , let $A = \text{Submatrix}(H, 1, 1, n - 1, n - 1)$. Let $e_t \in \mathbb{F}_2^{n-1}$ with $e_t[t] = 1$, and $e_t[j] = 0$ for $j \neq t$. Let $S_{(\lambda_1, \dots, \lambda_{n-1})}^1 = \mathbb{F}_{2^n} \setminus \text{Span}(\lambda_1 A[1] + \dots + \lambda_{n-1} A[n-1])$ for all $(\lambda_1, \lambda_2, \dots, \lambda_{n-1}) \in \mathbb{F}_2^{n-1} \setminus \{0\}$. Let $i = 1$.

Step 2. For each $x_i \in S_{e_i}^i$, do Step 3.

Step 3. If $i = n - 1$, then do Step 5, else do Step 4.

Step 4. Let $H[i, n] = H[n, i] = x_i$. For all $(\lambda_{i+1}, \dots, \lambda_{n-1}) \in \mathbb{F}_2^{n-1-i} \setminus \{0\}$, define $S_{\lambda}^{i+1} = S_{\lambda}^i \cap S_{\lambda \oplus e_i}^i$, where $\lambda = (0, \dots, 0, \lambda_{i+1}, \dots, \lambda_{n-1}) \in \mathbb{F}_2^{n-1}$. Then let $i := i + 1$, turn to Step 2.

Step 5. Let $H[n-1, n] = H[n, n-1] = x_{n-1}$, then output H .

Example 1: Algorithm 1 is the core part of our program. If we want to find new APN functions on \mathbb{F}_{2^n} for $n \geq 8$, then we must change the values of a QAM for at least two columns (and rows). We give an example in the following. It is well-known that x^3 is a restricted quadratic APN function on \mathbb{F}_{2^n} . Let $n = 8$, g be the default primitive element used in Magma, and C be an 8×8 matrix such that $C[1, 2] = C[2, 1] = 1$ and $C[i, t] = 0$ for all the other values. Suppose M is an 8×8 matrix such that $M[i, j] = (g^{11})^{2^{i-1}+2^{j-1}}$ for $1 \leq i, j \leq n$ (Note that $\{(g^{11}), (g^{11})^2, \dots, (g^{11})^{2^{n-1}}\}$ is a basis of \mathbb{F}_{2^8} over \mathbb{F}_2). Then we can get the corresponding QAM of x^3 : $H = M^t CM$ =

$$\begin{pmatrix} 0 & g^{34} & g^{81} & g^{83} & g^{170} & g^{106} & g^{84} & g^{17} \\ g^{34} & 0 & g^{68} & g^{162} & g^{166} & g^{85} & g^{212} & g^{168} \\ g^{81} & g^{68} & 0 & g^{136} & g^{69} & g^{77} & g^{170} & g^{169} \\ g^{83} & g^{162} & g^{136} & 0 & g^{17} & g^{138} & g^{154} & g^{85} \\ g^{170} & g^{166} & g^{69} & g^{17} & 0 & g^{34} & g^{21} & g^{53} \\ g^{106} & g^{85} & g^{77} & g^{138} & g^{34} & 0 & g^{68} & g^{42} \\ g^{84} & g^{212} & g^{170} & g^{154} & g^{21} & g^{68} & 0 & g^{136} \\ g^{17} & g^{168} & g^{169} & g^{85} & g^{53} & g^{42} & g^{136} & 0 \end{pmatrix}.$$

Reassign the values of the last two columns (and rows), we

can get a new QAM $H' =$

$$\begin{pmatrix} 0 & g^{34} & g^{81} & g^{83} & g^{170} & g^{106} & g^{84} & 1 \\ g^{34} & 0 & g^{68} & g^{162} & g^{166} & g^{85} & g^{212} & g^{233} \\ g^{81} & g^{68} & 0 & g^{136} & g^{69} & g^{77} & g^{170} & g^{165} \\ g^{83} & g^{162} & g^{136} & 0 & g^{17} & g^{138} & g^{64} & g^{68} \\ g^{170} & g^{166} & g^{69} & g^{17} & 0 & g^{34} & g^{235} & g^{250} \\ g^{106} & g^{85} & g^{77} & g^{138} & g^{34} & 0 & g^{151} & g^{81} \\ g^{84} & g^{212} & g^{170} & g^{64} & g^{235} & g^{151} & 0 & g^{113} \\ 1 & g^{233} & g^{165} & g^{68} & g^{250} & g^{81} & g^{113} & 0 \end{pmatrix}.$$

The corresponding APN function of H' is $F'(x) = g^{145} * x^{192} + g^{173} * x^{160} + g^{239} * x^{144} + g^{141} * x^{136} + g^{197} * x^{132} + g^{35} * x^{130} + g^{92} * x^{129} + g^7 * x^{96} + g^{176} * x^{80} + g^{99} * x^{72} + g^{135} * x^{68} + g^{182} * x^{66} + g^{34} * x^{65} + g^{117} * x^{48} + g^{36} * x^{40} + g^{108} * x^{36} + g^{160} * x^{34} + g^{187} * x^{33} + g^3 * x^{24} + g^{27} * x^{20} + g^{156} * x^{18} + g^{215} * x^{17} + g^{99} * x^{12} + g^{188} * x^{10} + g * x^9 + g^{137} * x^6 + g^{225} * x^5 + g^{206} * x^3$, which is CCZ-inequivalent to x^3 . So we have got a new APN functions over \mathbb{F}_{2^8} .

C. Experimental results

We have implemented the algorithm in this paper. In this subsection we will report experiment results using our algorithm.

(i) Dillon [12] listed 18 classes of CCZ-inequivalent APN functions over \mathbb{F}_{2^7} . Edel [16] found a new class of APN function and this list expanded to 19 classes. With our method, firstly we can construct a 7×7 QAM H from x^3 , then reassign the values $H[3, 6], H[3, 7], H[4, 5], H[4, 6], H[4, 7], H[5, 6], H[5, 7]$ and $H[6, 7]$ (during this process, we must keep H symmetric). Using this idea we can get more than 470 classes of CCZ-inequivalent quadratic APN functions, and these functions are all CCZ-inequivalent to the known ones. Similar method can be used on \mathbb{F}_{2^6} . According to Edel's results [14], there is only 13 classes of CCZ-inequivalent quadratic APN functions. Our program shows that it is only need to change 8 (2×4) elements of of a QAM and get all the 13 classes of CCZ-inequivalent quadratic APN functions. This method is an early version of our program, which works efficiently on \mathbb{F}_{2^n} for $n \leq 7$. But when $n \geq 8$, it becomes very slow.

(ii) For now, the algorithm based on solving Problem 1 is the most efficient method for finding new QAMs. It should be noted that we must change the last two columns (and rows) of a known QAM to get new QAMs when $n \geq 8$. Algorithm 1 can be implemented in parallel, and we are running our programs in many computers now. Up to now, we have found more than 1000 classes of CCZ-inequivalent quadratic APN functions on \mathbb{F}_{2^8} , and they are all CCZ-inequivalent to the 23 classes of known ones introduced by Dillon [12] and Edel [16]. We have checked all these new APN functions with the method introduced in [4], none of them is CCZ-equivalent to a permutation.

V. CONCLUSION

We find a correspondence relationship between quadratic APN functions and a special kind of matrices. Based on this result, we have developed efficient algorithms to construct

quadratic APN functions. Up to now, we have found more than 470 and 1000 classes of new CCZ-inequivalent quadratic APN functions on \mathbb{F}_{2^7} and \mathbb{F}_{2^8} respectively (See the appendices). We think our lists are not complete, especially the list on \mathbb{F}_{2^8} . It is far from complete, so we will add some new quadratic APN functions in the lists in the future. Certainly we will also check whether the new APN functions are CCZ-equivalent to some permutations. Our idea is just a beginning. Much related work can be done in the future, such as, finding some QAMs whose corresponding functions are APN on \mathbb{F}_{2^n} for infinite many n , generalizing the matrix approach to construct PN functions, and finding better methods to construct QAMs, etc.

REFERENCES

- [1] C. Bracken, E. Byrne, N. Markin, and G. McGuire, "New families of quadratic almost perfect nonlinear trinomials and multinomials," *Finite Fields and Their Appl.*, vol. 14, no. 3, pp. 703-714, 2008.
- [2] C. Bracken, E. Byrne, N. Markin, and G. McGuire, "A few more quadratic APN functions," *Cryptogr. Commun.*, vol. 3, no. 3, pp. 43-53, 2011.
- [3] K. Browning, J. F. Dillon, and M. McQuistan, "APN polynomials and related codes," Special volume of *Journal of Combinatorics, Information and System Sciences*, honoring the 75-th birthday of Prof. D.K.Ray-Chaudhuri, vol. 34, no. 1-4, pp. 135-159, 2009.
- [4] K. Browning, J. F. Dillon, M. T. McQuistan and A. J. Wolfe, "An APN permutation in dimension six," *Contemporary Mathematics*, vol. 58, pp. 33-42, 2010.
- [5] L. Budaghyan, C. Carlet and A. Pott, "New classes of almost bent and almost perfect nonlinear polynomials," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1141-1152, 2006.
- [6] L. Budaghyan and C. Carlet, "Classes of quadratic APN trinomials and hexanomials and related structures," *IEEE Trans. Inf. Theory*, vol. 54, no. 5, pp. 2354-2357, 2008.
- [7] L. Budaghyan, C. Carlet and G. Leander, "Constructing new APN functions from known ones," *Finite Fields and Their Appl.*, vol. 15, no. 2, pp. 150-159, 2009.
- [8] L. Budaghyan, C. Carlet and G. Leander, "Two classes of quadratic APN binomials inequivalent to power functions," *IEEE Trans. Inf. Theory*, vol. 54, no. 9, pp. 4218-4229, 2008.
- [9] C. Carlet. *Vectorial Boolean Functions for Cryptography*, In: Crama, Y., Hammer, P. (eds.) *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, pp. 398-469. Cambridge University Press, Cambridge (2010); Preliminary version available at <http://www-rocq.inria.fr/codes/Claude.Carlet/pubs.html>.
- [10] C. Carlet, P. Charpin, and V. Zinoviev, "Codes, bent functions and permutations suitable for DES-like cryptosystems," *Designs, Codes and Cryptography*, 15(2):125-156, 1998.
- [11] J. Daemen and V. Rijmen, *AES Proposal: Rijndael*, 1999, [Online]. Available: <http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf>.
- [12] J. F. Dillon, "APN polynomials: an update," *Fq9*, The 9th International Conference on Finite Fields and Appl., Dublin, Ireland, 2009.
- [13] Y. Edel, "Geometrical and combinatorial aspects of APN functions", Contact Forum: Coding Theory and Cryptography III, Brussels, (2009) [Online]. Available: <http://cage.ugent.be/~ls/website2009/abstracts/slidesyvesedel.pdf>.
- [14] Y. Edel, "Quadratic APN functions as subspaces of alternating bilinear forms," *Proceedings of the Contact Forum Coding Theory and Cryptography III*, Belgium (2009), pp. 11-24, 2011.
- [15] Y. Edel, G. Kyureghyan, and A. Pott, "A new APN function which is not equivalent to a power mapping," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 744-747, 2006.
- [16] Y. Edel and A. Pott, "A new almost perfect nonlinear function which is not quadratic," *Adv. Math. Commun.* vol. 3, no. 1, pp. 59-81, 2009.
- [17] R. Gold, "Maximal recursive sequences with 3-valued recursive cross-correlation functions," *IEEE Trans. Inf. Theory*, vol. 14, no. 1, pp. 154-156, 1968.
- [18] R. Lidl and H. Niederreiter, *Finite fields*. Cambridge, U.K.: Cambridge Univ. Press, pp. 58, 1983.
- [19] S. Ling and L. Qu, "A note on linearized polynomials and the dimension of their kernels," *Finite Fields and Their Appl.*, vol. 18, no. 1, pp. 56-62, 2012.
- [20] K. Nyberg and L. R. Knudsen, "Provable security against differential cryptanalysis", *CRYPTO 92*, LNCS 740 Springer-Verlag, 566-574.
- [21] S. Yoshiara, "Equivalences of quadratic APN functions," *J. Algebr. Comb.*, vol. 35, no. 3, pp. 461-475, 2012.
- [22] Z. Zha and X. Wang, "Almost Perfect Nonlinear Power Functions in Odd Characteristic," *IEEE Trans. Inf. Theory*, vol. 57, no. 7, pp. 4826-4832, 2011.

Yuyin Yu received the B.S. degree in computer science from Henan Normal University, Henan, China. At present he is a Ph.D. candidate at Institute of Infomation Engineering, Chinese Academy of Sciences, Beijing. His research interests include APN functions, cryptographic algorithms and related areas.

Mingsheng Wang received the Ph.D. degree in 1994 from Beijing Normal University, Beijing, China. At present he is a research professor at Institute of Information Engineering, Chinese Academy of Sciences, Beijing. His research interests lie in computational algebra, cryptography, and information security.

Yongqiang Li received B.S. and M.S. degrees in Mathematics from Beijing Normal University, Beijing, China, in 2005 and 2008 respectively and the Ph.D. degree in Information Security from Institute of Software, Chinese Academy of Sciences, China, in January 2012. At present he is an assistant professor at Institute of Information Engineering, Chinese Academy of Sciences. His research interests lie in cryptography and related areas.

1 Appendix 1: 490 CCZ-inequivalent APN functions on \mathbb{F}_{2^7}

g is the default primitive element of \mathbb{F}_{2^7} in Magma (V2.12-16).

```

x^3,
x^9,
x^5,
x^13,
x^57,
x^126,
x^3 + x^9 + x^10 + x^66 + x^80,
x^5 + x^18 + x^34,
x^3 + x^6 + x^20,
x^3 + x^17 + x^33 + x^34,
x^3 + x^5 + x^10 + x^33 + x^34,
x^3 + x^9 + x^18 + x^66,
x^3 + x^12 + x^17 + x^33,
x^3 + x^20 + x^34 + x^66,
x^3 + x^12 + x^40 + x^72,
x^3 + x^6 + x^34 + x^40 + x^72,
x^3 + x^5 + x^6 + x^12 + x^33 + x^34,
x^32*x^96 + g^7*x^80 + g^22*x^72 + g^60*x^68 + g^37*x^66 + g^93*x^48 + g^43*x^40 + g^37*x^36 + g^114*x^33 + g^60*x^24 + g^22*x^20 + g^32*x^18 + g^67*x^12 + g^107*x^10 + g^93*x^9 + g^64*x^6 + g^121*x^5 + g^22*x^3,
g^27*x^96 + g^74*x^80 + g^108*x^66 + g^90*x^48 + g^15*x^40 + g^84*x^36 + g^126*x^34 + g*x^33 + g^103*x^24 + g^105*x^18 + g^118*x^12 + g^119*x^10 + g^24*x^6 + g^23*x^3,
g^3*x^96 + g^126*x^80 + g^116*x^72 + g^36*x^68 + g^109*x^66 + g^108*x^48 + g^76*x^34 + g^115*x^33 + g^79*x^24 + g^10*x^20 + g^80*x^18 + g^117*x^17 + g^4*x^10 + g^62*x^9 + g^39*x^6 + g^102*x^5 + g^26*x^3,
g^95*x^96 + g^112*x^68 + g^93*x^66 + g^30*x^48 + g^103*x^36 + g^44*x^34 + g^77*x^33 + g^50*x^24 + g^118*x^20 + g^20*x^18 + g^101*x^17 + g^91*x^12 + g^13*x^10 + g^115*x^9 + g^33*x^6 + g^55*x^3,
g^49*x^96 + g^97*x^80 + g^100*x^72 + g*x^68 + g^42*x^66 + g^21*x^65 + g^29*x^48 + g^122*x^36 + g^2*x^34 + g^117*x^33 + g^68*x^24 + g^74*x^20 + x^18 + g^13*x^17 + g^95*x^12 + g^90*x^10 + g^48*x^9 + g^55*x^6 + g^79*x^5 + g^124*x^3,
g^10*x^96 + g^123*x^66 + g^117*x^65 + g^17*x^48 + g^52*x^40 + g^3*x^36 + g^40*x^34 + g^95*x^33 + g^57*x^24 + g^5*x^20 + g^122*x^18 + g^105*x^17 + g^39*x^12 + g^41*x^10 + g^33*x^9 + g^73*x^6 + g^9*x^5 + g^10*x^3,
g^54*x^96 + g^80*x^80 + g^54*x^72 + g^39*x^68 + x^66 + g^28*x^40 + g^9*x^36 + g^46*x^33 + g^2*x^24 + g^76*x^20 + g^125*x^18 + g^66*x^12 + g^60*x^10 + g^36*x^9 + g^90*x^6 + g^60*x^5 + g^121*x^3,
g^118*x^96 + g^83*x^80 + g^70*x^72 + g^20*x^68 + g^126*x^66 + g^100*x^65 + g^3*x^34 + g^15*x^33 + g^2*x^24 + g^78*x^20 + g^123*x^18 + g^58*x^17 + g^15*x^12 + g^80*x^9 + g^15*x^6 + g^72*x^3,
g^6*x^96 + g^121*x^80 + g^26*x^68 + g^121*x^66 + g^82*x^65 + g^123*x^64 + g^17*x^48 + g^52*x^40 + g^9*x^36 + g^46*x^33 + g^2*x^24 + g^76*x^20 + g^125*x^18 + g^66*x^12 + g^60*x^10 + g^36*x^9 + g^90*x^6 + g^60*x^5 + g^121*x^3,
g^77*x^96 + g^85*x^80 + g^36*x^72 + g^72*x^68 + g^55*x^68 + g^42*x^65 + g^79*x^48 + g^73*x^40 + g^38*x^36 + g^66*x^34 + g^4*x^33 + g^108*x^24 + g^40*x^20 + g^74*x^18 + g^32*x^17 + g^36*x^9 + g^94*x^10 + g^32*x^8 + g^80*x^6 + g^2*x^5 + g^81*x^3,
g^69*x^96 + g^110*x^80 + g^100*x^72 + g^90*x^68 + g^91*x^66 + g^6*x^65 + g^41*x^48 + g^43*x^34 + g^111*x^33 + g^23*x^24 + g^9*x^20 + g^67*x^18 + g^123*x^12 + g^68*x^9 + g^122*x^6 + g^31*x^5 + g^99*x^3,
g^54*x^96 + g^56*x^80 + g^54*x^72 + g^56*x^68 + g^54*x^66 + g^54*x^64 + g^54*x^62 + g^54*x^60 + g^54*x^58 + g^54*x^56 + g^54*x^54 + g^54*x^52 + g^54*x^50 + g^54*x^48 + g^54*x^46 + g^54*x^44 + g^54*x^42 + g^54*x^40 + g^54*x^38 + g^54*x^36 + g^54*x^34 + g^54*x^32 + g^54*x^30 + g^54*x^28 + g^54*x^26 + g^54*x^24 + g^54*x^22 + g^54*x^20 + g^54*x^18 + g^54*x^16 + g^54*x^14 + g^54*x^12 + g^54*x^10 + g^54*x^8 + g^54*x^6 + g^54*x^4 + g^54*x^2 + g^54*x^0 + g^54*x^1 + g^54*x^3 + g^54*x^5 + g^54*x^7 + g^54*x^9 + g^54*x^11 + g^54*x^13 + g^54*x^15 + g^54*x^17 + g^54*x^19 + g^54*x^21 + g^54*x^23 + g^54*x^25 + g^54*x^27 + g^54*x^29 + g^54*x^31 + g^54*x^33 + g^54*x^35 + g^54*x^37 + g^54*x^39 + g^54*x^41 + g^54*x^43 + g^54*x^45 + g^54*x^47 + g^54*x^49 + g^54*x^51 + g^54*x^53 + g^54*x^55 + g^54*x^57 + g^54*x^59 + g^54*x^61 + g^54*x^63 + g^54*x^65 + g^54*x^67 + g^54*x^69 + g^54*x^71 + g^54*x^73 + g^54*x^75 + g^54*x^77 + g^54*x^79 + g^54*x^81 + g^54*x^83 + g^54*x^85 + g^54*x^87 + g^54*x^89 + g^54*x^91 + g^54*x^93 + g^54*x^95 + g^54*x^97 + g^54*x^99 + g^54*x^101 + g^54*x^103 + g^54*x^105 + g^54*x^107 + g^54*x^109 + g^54*x^111 + g^54*x^113 + g^54*x^115 + g^54*x^117 + g^54*x^119 + g^54*x^121 + g^54*x^123 + g^54*x^125 + g^54*x^127 + g^54*x^129 + g^54*x^131 + g^54*x^133 + g^54*x^135 + g^54*x^137 + g^54*x^139 + g^54*x^141 + g^54*x^143 + g^54*x^145 + g^54*x^147 + g^54*x^149 + g^54*x^151 + g^54*x^153 + g^54*x^155 + g^54*x^157 + g^54*x^159 + g^54*x^161 + g^54*x^163 + g^54*x^165 + g^54*x^167 + g^54*x^169 + g^54*x^171 + g^54*x^173 + g^54*x^175 + g^54*x^177 + g^54*x^179 + g^54*x^181 + g^54*x^183 + g^54*x^185 + g^54*x^187 + g^54*x^189 + g^54*x^191 + g^54*x^193 + g^54*x^195 + g^54*x^197 + g^54*x^199 + g^54*x^201 + g^54*x^203 + g^54*x^205 + g^54*x^207 + g^54*x^209 + g^54*x^211 + g^54*x^213 + g^54*x^215 + g^54*x^217 + g^54*x^219 + g^54*x^221 + g^54*x^223 + g^54*x^225 + g^54*x^227 + g^54*x^229 + g^54*x^231 + g^54*x^233 + g^54*x^235 + g^54*x^237 + g^54*x^239 + g^54*x^241 + g^54*x^243 + g^54*x^245 + g^54*x^247 + g^54*x^249 + g^54*x^251 + g^54*x^253 + g^54*x^255 + g^54*x^257 + g^54*x^259 + g^54*x^261 + g^54*x^263 + g^54*x^265 + g^54*x^267 + g^54*x^269 + g^54*x^271 + g^54*x^273 + g^54*x^275 + g^54*x^277 + g^54*x^279 + g^54*x^281 + g^54*x^283 + g^54*x^285 + g^54*x^287 + g^54*x^289 + g^54*x^291 + g^54*x^293 + g^54*x^295 + g^54*x^297 + g^54*x^299 + g^54*x^301 + g^54*x^303 + g^54*x^305 + g^54*x^307 + g^54*x^309 + g^54*x^311 + g^54*x^313 + g^54*x^315 + g^54*x^317 + g^54*x^319 + g^54*x^321 + g^54*x^323 + g^54*x^325 + g^54*x^327 + g^54*x^329 + g^54*x^331 + g^54*x^333 + g^54*x^335 + g^54*x^337 + g^54*x^339 + g^54*x^341 + g^54*x^343 + g^54*x^345 + g^54*x^347 + g^54*x^349 + g^54*x^351 + g^54*x^353 + g^54*x^355 + g^54*x^357 + g^54*x^359 + g^54*x^361 + g^54*x^363 + g^54*x^365 + g^54*x^367 + g^54*x^369 + g^54*x^371 + g^54*x^373 + g^54*x^375 + g^54*x^377 + g^54*x^379 + g^54*x^381 + g^54*x^383 + g^54*x^385 + g^54*x^387 + g^54*x^389 + g^54*x^391 + g^54*x^393 + g^54*x^395 + g^54*x^397 + g^54*x^399 + g^54*x^401 + g^54*x^403 + g^54*x^405 + g^54*x^407 + g^54*x^409 + g^54*x^411 + g^54*x^413 + g^54*x^415 + g^54*x^417 + g^54*x^419 + g^54*x^421 + g^54*x^423 + g^54*x^425 + g^54*x^427 + g^54*x^429 + g^54*x^431 + g^54*x^433 + g^54*x^435 + g^54*x^437 + g^54*x^439 + g^54*x^441 + g^54*x^443 + g^54*x^445 + g^54*x^447 + g^54*x^449 + g^54*x^451 + g^54*x^453 + g^54*x^455 + g^54*x^457 + g^54*x^459 + g^54*x^461 + g^54*x^463 + g^54*x^465 + g^54*x^467 + g^54*x^469 + g^54*x^471 + g^54*x^473 + g^54*x^475 + g^54*x^477 + g^54*x^479 + g^54*x^481 + g^54*x^483 + g^54*x^485 + g^54*x^487 + g^54*x^489 + g^54*x^491 + g^54*x^493 + g^54*x^495 + g^54*x^497 + g^54*x^499 + g^54*x^501 + g^54*x^503 + g^54*x^505 + g^54*x^507 + g^54*x^509 + g^54*x^511 + g^54*x^513 + g^54*x^515 + g^54*x^517 + g^54*x^519 + g^54*x^521 + g^54*x^523 + g^54*x^525 + g^54*x^527 + g^54*x^529 + g^54*x^531 + g^54*x^533 + g^54*x^535 + g^54*x^537 + g^54*x^539 + g^54*x^541 + g^54*x^543 + g^54*x^545 + g^54*x^547 + g^54*x^549 + g^54*x^551 + g^54*x^553 + g^54*x^555 + g^54*x^557 + g^54*x^559 + g^54*x^561 + g^54*x^563 + g^54*x^565 + g^54*x^567 + g^54*x^569 + g^54*x^571 + g^54*x^573 + g^54*x^575 + g^54*x^577 + g^54*x^579 + g^54*x^581 + g^54*x^583 + g^54*x^585 + g^54*x^587 + g^54*x^589 + g^54*x^591 + g^54*x^593 + g^54*x^595 + g^54*x^597 + g^54*x^599 + g^54*x^601 + g^54*x^603 + g^54*x^605 + g^54*x^607 + g^54*x^609 + g^54*x^611 + g^54*x^613 + g^54*x^615 + g^54*x^617 + g^54*x^619 + g^54*x^621 + g^54*x^623 + g^54*x^625 + g^54*x^627 + g^54*x^629 + g^54*x^631 + g^54*x^633 + g^54*x^635 + g^54*x^637 + g^54*x^639 + g^54*x^641 + g^54*x^643 + g^54*x^645 + g^54*x^647 + g^54*x^649 + g^54*x^651 + g^54*x^653 + g^54*x^655 + g^54*x^657 + g^54*x^659 + g^54*x^661 + g^54*x^663 + g^54*x^665 + g^54*x^667 + g^54*x^669 + g^54*x^671 + g^54*x^673 + g^54*x^675 + g^54*x^677 + g^54*x^679 + g^54*x^681 + g^54*x^683 + g^54*x^685 + g^54*x^687 + g^54*x^689 + g^54*x^691 + g^54*x^693 + g^54*x^695 + g^54*x^697 + g^54*x^699 + g^54*x^701 + g^54*x^703 + g^54*x^705 + g^54*x^707 + g^54*x^709 + g^54*x^711 + g^54*x^713 + g^54*x^715 + g^54*x^717 + g^54*x^719 + g^54*x^721 + g^54*x^723 + g^54*x^725 + g^54*x^727 + g^54*x^729 + g^54*x^731 + g^54*x^733 + g^54*x^735 + g^54*x^737 + g^54*x^739 + g^54*x^741 + g^54*x^743 + g^54*x^745 + g^54*x^747 + g^54*x^749 + g^54*x^751 + g^54*x^753 + g^54*x^755 + g^54*x^757 + g^54*x^759 + g^54*x^761 + g^54*x^763 + g^54*x^765 + g^54*x^767 + g^54*x^769 + g^54*x^771 + g^54*x^773 + g^54*x^775 + g^54*x^777 + g^54*x^779 + g^54*x^781 + g^54*x^783 + g^54*x^785 + g^54*x^787 + g^54*x^789 + g^54*x^791 + g^54*x^793 + g^54*x^795 + g^54*x^797 + g^54*x^799 + g^54*x^801 + g^54*x^803 + g^54*x^805 + g^54*x^807 + g^54*x^809 + g^54*x^811 + g^54*x^813 + g^54*x^815 + g^54*x^817 + g^54*x^819 + g^54*x^821 + g^54*x^823 + g^54*x^825 + g^54*x^827 + g^54*x^829 + g^54*x^831 + g^54*x^833 + g^54*x^835 + g^54*x^837 + g^54*x^839 + g^54*x^841 + g^54*x^843 + g^54*x^845 + g^54*x^847 + g^54*x^849 + g^54*x^851 + g^54*x^853 + g^54*x^855 + g^54*x^857 + g^54*x^859 + g^54*x^861 + g^54*x^863 + g^54*x^865 + g^54*x^867 + g^54*x^869 + g^54*x^871 + g^54*x^873 + g^54*x^875 + g^54*x^877 + g^54*x^879 + g^54*x^881 + g^54*x^883 + g^54*x^885 + g^54*x^887 + g^54*x^889 + g^54*x^891 + g^54*x^893 + g^54*x^895 + g^54*x^897 + g^54*x^899 + g^54*x^901 + g^54*x^903 + g^54*x^905 + g^54*x^907 + g^54*x^909 + g^54*x^911 + g^54*x^913 + g^54*x^915 + g^54*x^917 + g^54*x^919 + g^54*x^921 + g^54*x^923 + g^54*x^925 + g^54*x^927 + g^54*x^929 + g^54*x^931 + g^54*x^933 + g^54*x^935 + g^54*x^937 + g^54*x^939 + g^54*x^941 + g^54*x^943 + g^54*x^945 + g^54*x^947 + g^54*x^949 + g^54*x^951 + g^54*x^953 + g^54*x^955 + g^54*x^957 + g^54*x^959 + g^54*x^961 + g^54*x^963 + g^54*x^965 + g^54*x^967 + g^54*x^969 + g^54*x^971 + g^54*x^973 + g^54*x^975 + g^54*x^977 + g^54*x^979 + g^54*x^981 + g^54*x^983 + g^54*x^985 + g^54*x^987 + g^54*x^989 + g^54*x^991 + g^54*x^993 + g^54*x^995 + g^54*x^997 + g^54*x^999 + g^54*x^1001 + g^54*x^1003 + g^54*x^1005 + g^54*x^1007 + g^54*x^1009 + g^54*x^1011 + g^54*x^1013 + g^54*x^1015 + g^54*x^1017 + g^54*x^1019 + g^54*x^1021 + g^54*x^1023 + g^54*x^1025 + g^54*x^1027 + g^54*x^1029 + g^54*x^1031 + g^54*x^1033 + g^54*x^1035 + g^54*x^1037 + g^54*x^1039 + g^54*x^1041 + g^54*x^1043 + g^54*x^1045 + g^54*x^1047 + g^54*x^1049 + g^54*x^1051 + g^54*x^1053 + g^54*x^1055 + g^54*x^1057 + g^54*x^1059 + g^54*x^1061 + g^54*x^1063 + g^54*x^1065 + g^54*x^1067 + g^54*x^1069 + g^54*x^10
```


$g^{121*x^96} + g^{85*x^80} + x^{72} + g^{107*x^{68}} + g^{93*x^{66}} + g^{117*x^{65}} + g^{85*x^{48}} + g^{116*x^{40}} + g^{82*x^{36}} + g^{29*x^{33}} + g^{95*x^{24}} + g^{31*x^{20}} + g^{60*x^{18}} + g^{119*x^{17}} + g^{45*x^{12}} + g^{29*x^{10}} + g^{83*x^9} + g^{109*x^6} + g^{73*x^5} + g^{81*x^3},$
 $g^{14*x^96} + g^{111*x^{80}} + g^{72*x^{72}} + g^{93*x^{68}} + g^{83*x^{66}} + g^{77*x^{65}} + g^{111*x^{48}} + g^{101*x^{40}} + g^{88*x^{36}} + g^{81*x^{34}} + g^{43*x^{33}} + g^{47*x^{24}} + g^{74*x^{20}} + g^{123*x^{18}} + g^{82*x^{17}} + g^{60*x^{12}} + g^{101*x^{10}} + g^{118*x^9} + g^{64*x^6} + g^{63*x^5} + g^{57*x^3},$
 $g^{74*x^96} + g^{78*x^{80}} + g^{99*x^{72}} + g^{54*x^{68}} + g^{116*x^{66}} + g^{97*x^{65}} + g^{110*x^{48}} + g^{91*x^{40}} + g^{41*x^{36}} + g^{122*x^{34}} + g^{37*x^{33}} + g^{101*x^{20}} + g^{97*x^{19}} + g^{20*x^{17}} + g^{53*x^{18}} + g^{10*x^{12}} + g^{102*x^{10}} + g^{110*x^9} + g^{67*x^6} + g^{106*x^5} + g^{79*x^3},$
 $g^{37*x^{96}} + g^{108*x^{80}} + g^{106*x^{72}} + g^{83*x^{68}} + g^{103*x^{66}} + g^{40*x^{40}} + g^{74*x^{36}} + g^{6*x^{34}} + g^{11*x^{33}} + g^{88*x^{24}} + g^{100*x^{18}} + g^{95*x^{17}} + g^{97*x^{12}} + g^{92*x^{10}} + g^{2*x^{9}} + g^{81*x^5} + g^{84*x^3},$
 $g^{84*x^96} + g^{12*x^{80}} + g^{16*x^{72}} + g^{110*x^{68}} + g^{60*x^{66}} + g^{109*x^{48}} + g^{84*x^{40}} + g^{61*x^{36}} + g^{126*x^{34}} + g^{70*x^{33}} + g^{85*x^{24}} + g^{8*x^{20}} + g^{59*x^{18}} + g^{94*x^{17}} + g^{34*x^{12}} + g^{105*x^{10}} + g^{99*x^9} + g^{41*x^6} + g^{108*x^5} + g*x^3, \dots$

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

2 Appendix 2: 1041 CCZ-inequivalent APN functions on \mathbb{F}_{2^8}

g is the default primitive element of \mathbb{F}_{2^8} in Magma (V2.12-16).

x³,
x⁹,
x⁵⁷,
g^{15*x^48 + g^16*x^33 + g^16*x^18 + x^17 + x^3},
x^{72 + x^6 + x^3},
x^{144 + x^6 + x^3},
g^{21*x^144 + g^183*x^66 + g^245*x^33 + x^3},
g^{135*x^144 + g^120*x^66 + g^65*x^18 + x^3},
g^{67*x^192 + g^182*x^132 + g^24*x^6 + x^3},
x^{160 + x^132 + x^80 + x^68 + x^6 + x^3},
x^{66 + x^40 + x^18 + x^5 + x^3},
x^{130 + x^66 + x^40 + x^12 + x^3},
g^{189*x^192 + g^143*x^144 + g^22*x^132 + g^21*x^129 + g^133*x^96 + g^239*x^66 + g^31*x^48 + g^187*x^36 + g^185*x^24 + g^168*x^18 + g^75*x^12 + g^91*x^9 + g^160*x^3},
g^{100*x^192 + g^12*x^160 + g^15*x^144 + g^243*x^136 + g^234*x^132 + g^33*x^130 + g^39*x^129 + g^139*x^96 + g^51*x^80 + g^229*x^72 + g^39*x^68 + g^17*x^66 + g^189*x^40 + g^198*x^36 + g^192*x^34 + g^217*x^33 + g^122*x^24 + g^144*x^20 + g^169*x^18 + g^141*x^17 + g^236*x^12 + g^117*x^10 + g^183*x^9 + g^184*x^6 + g^231*x^5 + g^228*x^3},
g^{155*x^192 + g^96*x^144 + g^223*x^132 + g^77*x^129 + g^69*x^66 + g^142*x^48 + g^168*x^36 + x^33 + g^145*x^24 + g^234*x^18 + g^202*x^12 + g^94*x^9 + g^189*x^6 + g^241*x^3},
g^{126*x^192 + g^119*x^144 + g^221*x^132 + g^222*x^129 + g^79*x^96 + g^221*x^72 + g^187*x^66 + g^148*x^48 + g^187*x^36 + g^237*x^24 + g^21*x^12 + g^119*x^9 + g^244*x^6 + g^236*x^3},
g^{151*x^192 + g^13*x^144 + g^58*x^132 + g^143*x^129 + g^110*x^96 + g^gx^72 + g^244*x^66 + g^26*x^48 + g^180*x^36 + g^8*x^33 + g^69*x^24 + g^76*x^18 + g^201*x^9 + g^19*x^6 + g^107*x^3},
g^{86*x^192 + g^224*x^129 + g^163*x^96 + g^102*x^66 + g^129*x^48 + g^102*x^33 + g^14*x^24 + g^170*x^18 + g^101*x^12 + g^58*x^6 + g^254*x^3},
g^{95*x^192 + g^242*x^144 + g^195*x^132 + g^98*x^129 + g^84*x^96 + g^45*x^72 + g^234*x^66 + g^202*x^48 + g^159*x^36 + g^58*x^33 + g^232*x^9 + g^54*x^6 + g^41*x^3},
g^{132*x^192 + g^37*x^144 + g^91*x^132 + g^162*x^72 + g^46*x^66 + g^252*x^48 + g^42*x^36 + g^81*x^33 + g^83*x^24 + g^185*x^12 + g^163*x^9 + g^216*x^6 + g^181*x^3},
g^{91*x^192 + g^124*x^144 + g^214*x^129 + g^59*x^96 + g^172*x^72 + g^138*x^66 + g^163*x^36 + g^100*x^33 + g^32*x^24 + g^250*x^18 + g^45*x^12 + g^241*x^6 + g^157*x^3},
g^{25*x^192 + g^140*x^144 + g^59*x^132 + g^129*x^129 + g^42*x^96 + g^164*x^72 + g^149*x^66 + g^119*x^48 + g^74*x^36 + g^211*x^33 + g^9*x^24 + g^46*x^18 + g^130*x^12 + g^185*x^9 + g^147*x^6 + g^27*x^3},
g^{113*x^192 + g^56*x^144 + g^68*x^132 + g^155*x^129 + g^91*x^96 + g^78*x^72 + g^159*x^66 + g^30*x^48 + g^194*x^36 + g^14*x^33 + g^238*x^24 + g^91*x^18 + g^100*x^12 + g^96*x^9 + g^222*x^6 + g^178*x^3},
/23
g^{145*x^192 + g^173*x^160 + g^239*x^144 + g^141*x^136 + g^197*x^132 + g^92*x^129 + g^7*x^96 + g^176*x^80 + g^99*x^72 + g^135*x^65 + g^117*x^48 + g^36*x^40 + g^108*x^34 + g^3*x^24 + g^215*x^17 + g^188*x^10 + g^156*x^9 + g^225*x^5 + g^206*x^3},
g^{162*x^192 + g^206*x^160 + g^187*x^144 + g^225*x^136 + g^197*x^132 + g^120*x^130 + g^41*x^129 + g^221*x^96 + g^234*x^65 + g^93*x^48 + g^195*x^40 + g^109*x^20 + g^129*x^18 + g^197*x^10 + g^127*x^9 + g^6*x^6 + g^85*x^5 + g^145*x^3},
g^{125*x^192 + g^45*x^144 + g^90*x^136 + g^141*x^130 + g^90*x^129 + g^48*x^80 + g^104*x^72 + g^250*x^68 + g^244*x^65 + g^45*x^48 + g^225*x^40 + g^16*x^36 + g^67*x^33 + g^142*x^24 + g^92*x^20 + g^11*x^18 + g^gx^17 + g^82*x^12 + g^159*x^10 + g^171*x^9 + g^42*x^6 + g^121*x^5 + g^94*x^3},
g^{69*x^192 + g^233*x^160 + g^130*x^144 + g^74*x^136 + g^15*x^132 + g^187*x^96 + g^99*x^80 + g^218*x^72 + g^142*x^68 + g^201*x^65 + g^16*x^48 + g^37*x^40 + g^222*x^34 + g^114*x^33 + g^146*x^24 + g^119*x^20 + g^159*x^17 + g^248*x^12 + g^222*x^10 + g^235*x^6 + g^2*x^5 + g^213*x^3},
g^{118*x^192 + g^174*x^160 + g^141*x^144 + g^69*x^136 + g^219*x^132 + g^106*x^130 + g^250*x^129 + g^63*x^96 + g^147*x^80 + g^10*x^34 + g^231*x^34 + g^3*x^24 + g^17*x^20 + g^150*x^17 + g^212*x^12 + g^131*x^10 + g^150*x^9 + g^40*x^6 + g^40*x^5 + g^45*x^3},
g^{175*x^192 + g^19*x^160 + g^91*x^144 + g^24*x^136 + g^17*x^129 + g^7*x^72 + g^16*x^65 + g^14*x^48 + g^17*x^36 + g^15*x^24 + g^17*x^20 + g^15*x^17 + g^21*x^12 + g^13*x^10 + g^15*x^9 + g^42*x^6 + g^13*x^5 + g^17*x^3},
g^{151*x^192 + g^13*x^160 + g^10*x^144 + g^22*x^136 + g^15*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^24*x^136 + g^15*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^224*x^129 + g^163*x^96 + g^102*x^66 + g^129*x^48 + g^102*x^33 + g^14*x^24 + g^17*x^18 + g^101*x^12 + g^56*x^6 + g^254*x^3},
g^{95*x^192 + g^242*x^144 + g^195*x^132 + g^98*x^129 + g^84*x^96 + g^45*x^72 + g^234*x^66 + g^202*x^48 + g^159*x^36 + g^58*x^33 + g^232*x^9 + g^54*x^6 + g^25*x^3},
g^{132*x^192 + g^37*x^144 + g^91*x^132 + g^162*x^72 + g^46*x^66 + g^252*x^48 + g^42*x^36 + g^81*x^33 + g^83*x^24 + g^185*x^12 + g^163*x^9 + g^216*x^6 + g^181*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{91*x^192 + g^11*x^160 + g^8*x^144 + g^18*x^136 + g^13*x^129 + g^4*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{91*x^192 + g^11*x^160 + g^8*x^144 + g^18*x^136 + g^13*x^129 + g^4*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{91*x^192 + g^11*x^160 + g^8*x^144 + g^18*x^136 + g^13*x^129 + g^4*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{91*x^192 + g^11*x^160 + g^8*x^144 + g^18*x^136 + g^13*x^129 + g^4*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{91*x^192 + g^11*x^160 + g^8*x^144 + g^18*x^136 + g^13*x^129 + g^4*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{91*x^192 + g^11*x^160 + g^8*x^144 + g^18*x^136 + g^13*x^129 + g^4*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{91*x^192 + g^11*x^160 + g^8*x^144 + g^18*x^136 + g^13*x^129 + g^4*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{132*x^192 + g^41*x^160 + g^18*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g^{86*x^192 + g^18*x^160 + g^15*x^144 + g^23*x^136 + g^14*x^129 + g^5*x^72 + g^14*x^65 + g^12*x^48 + g^16*x^36 + g^14*x^24 + g^16*x^20 + g^14*x^17 + g^20*x^12 + g^12*x^10 + g^14*x^9 + g^30*x^6 + g^30*x^5 + g^35*x^3},
g

$g^{228*x^192} + g^{153*x^160} + g^{37*x^144} + g^{167*x^136} + g^{148*x^132} + g^{49*x^130} + g^{9*x^129} + g^{69*x^96} + g^{75*x^80} + g^{200*x^68} + g^{64*x^66} + g^{236*x^65} + g^{25*x^48} + g^{251*x^40} + g^{242*x^33} + g^{126*x^24} + g^{78*x^18} + g^{119*x^12} + g^{165*x^10} + g^{203*x^9} + g^{198*x^5} + g^{229*x^3},$
 $g^{165*x^192} + g^{33*x^160} + g^{247*x^144} + g^{100*x^136} + g^{7*x^132} + g^{115*x^130} + g^{117*x^129} + g^{154*x^96} + g^{22*x^80} + g^{198*x^72} + g^{171*x^68} + g^{174*x^65} + g^{207*x^48} + g^{24*x^33} + g^{162*x^24} + g^{233*x^20} + g^{14*x^18} + g^{9*x^12} + g^{21*x^6} + g^{245*x^3},$
 $g^{65*x^192} + g^{15*x^160} + g^{34*x^144} + g^{193*x^136} + g^{128*x^132} + g^{62*x^130} + g^{147*x^129} + g^{136*x^80} + g^{53*x^74} + g^{42*x^48} + g^{22*x^33} + g^{16*x^24} + g^{21*x^18} + g^{159*x^65} + g^{159*x^68} + g^{178*x^68} + g^{162*x^66} + g^{193*x^129} + g^{19*x^122} + g^{19*x^120} + g^{106*x^80} + g^{164*x^76} + g^{164*x^72} + g^{16*x^68} + g^{16*x^64} + g^{16*x^60} + g^{16*x^56} + g^{16*x^52} + g^{16*x^48} + g^{16*x^44} + g^{16*x^40} + g^{16*x^36} + g^{16*x^32} + g^{16*x^28} + g^{16*x^24} + g^{16*x^20} + g^{16*x^16} + g^{16*x^12} + g^{16*x^8} + g^{16*x^4} + g^{16*x^0},$
 $g^{21*x^192} + g^{2*x^160} + g^{145*x^136} + g^{113*x^132} + g^{23*x^130} + g^{25*x^128} + g^{21*x^126} + g^{23*x^124} + g^{21*x^122} + g^{23*x^120} + g^{21*x^118} + g^{23*x^116} + g^{21*x^114} + g^{23*x^112} + g^{21*x^110} + g^{23*x^108} + g^{21*x^106} + g^{23*x^104} + g^{21*x^102} + g^{23*x^100} + g^{21*x^98} + g^{23*x^96} + g^{21*x^94} + g^{23*x^92} + g^{21*x^90} + g^{23*x^88} + g^{21*x^86} + g^{23*x^84} + g^{21*x^82} + g^{23*x^80} + g^{21*x^78} + g^{23*x^76} + g^{21*x^74} + g^{23*x^72} + g^{21*x^70} + g^{23*x^68} + g^{21*x^66} + g^{23*x^64} + g^{21*x^62} + g^{23*x^60} + g^{21*x^58} + g^{23*x^56} + g^{21*x^54} + g^{23*x^52} + g^{21*x^50} + g^{23*x^48} + g^{21*x^46} + g^{23*x^44} + g^{21*x^42} + g^{23*x^40} + g^{21*x^38} + g^{23*x^36} + g^{21*x^34} + g^{23*x^32} + g^{21*x^30} + g^{23*x^28} + g^{21*x^26} + g^{23*x^24} + g^{21*x^22} + g^{23*x^20} + g^{21*x^18} + g^{23*x^16} + g^{21*x^14} + g^{23*x^12} + g^{21*x^10} + g^{23*x^8} + g^{21*x^6} + g^{23*x^4} + g^{21*x^2} + g^{23*x^0},$
 $g^{20*x^192} + g^{19*x^160} + g^{16*x^144} + g^{14*x^136} + g^{13*x^132} + g^{10*x^130} + g^{9*x^128} + g^{8*x^126} + g^{7*x^124} + g^{6*x^122} + g^{5*x^120} + g^{4*x^118} + g^{3*x^116} + g^{2*x^114} + g^{1*x^112} + g^{0*x^110},$
 $g^{19*x^192} + g^{18*x^160} + g^{15*x^144} + g^{13*x^136} + g^{12*x^132} + g^{10*x^130} + g^{9*x^128} + g^{8*x^126} + g^{7*x^124} + g^{6*x^122} + g^{5*x^120} + g^{4*x^118} + g^{3*x^116} + g^{2*x^114} + g^{1*x^112} + g^{0*x^110},$
 $g^{18*x^192} + g^{17*x^160} + g^{14*x^144} + g^{12*x^136} + g^{11*x^132} + g^{9*x^130} + g^{8*x^128} + g^{7*x^126} + g^{6*x^124} + g^{5*x^122} + g^{4*x^120} + g^{3*x^118} + g^{2*x^116} + g^{1*x^114} + g^{0*x^112},$
 $g^{17*x^192} + g^{16*x^160} + g^{13*x^144} + g^{11*x^136} + g^{10*x^132} + g^{8*x^130} + g^{7*x^128} + g^{6*x^126} + g^{5*x^124} + g^{4*x^122} + g^{3*x^120} + g^{2*x^118} + g^{1*x^116} + g^{0*x^114},$
 $g^{16*x^192} + g^{15*x^160} + g^{12*x^144} + g^{10*x^136} + g^{9*x^132} + g^{7*x^130} + g^{6*x^128} + g^{5*x^126} + g^{4*x^124} + g^{3*x^122} + g^{2*x^120} + g^{1*x^118} + g^{0*x^116},$
 $g^{15*x^192} + g^{14*x^160} + g^{11*x^144} + g^{9*x^136} + g^{8*x^132} + g^{6*x^130} + g^{5*x^128} + g^{4*x^126} + g^{3*x^124} + g^{2*x^122} + g^{1*x^120} + g^{0*x^118},$
 $g^{14*x^192} + g^{13*x^160} + g^{10*x^144} + g^{8*x^136} + g^{7*x^132} + g^{5*x^130} + g^{4*x^128} + g^{3*x^126} + g^{2*x^124} + g^{1*x^122} + g^{0*x^120},$
 $g^{13*x^192} + g^{12*x^160} + g^{9*x^144} + g^{7*x^136} + g^{6*x^132} + g^{4*x^130} + g^{3*x^128} + g^{2*x^126} + g^{1*x^124} + g^{0*x^122},$
 $g^{12*x^192} + g^{11*x^160} + g^{8*x^144} + g^{6*x^136} + g^{5*x^132} + g^{3*x^130} + g^{2*x^128} + g^{1*x^126} + g^{0*x^124},$
 $g^{11*x^192} + g^{10*x^160} + g^{7*x^144} + g^{5*x^136} + g^{4*x^132} + g^{2*x^130} + g^{1*x^128} + g^{0*x^126},$
 $g^{10*x^192} + g^{9*x^160} + g^{6*x^144} + g^{4*x^136} + g^{3*x^132} + g^{1*x^130} + g^{0*x^128},$
 $g^{9*x^192} + g^{8*x^160} + g^{5*x^144} + g^{3*x^136} + g^{2*x^132} + g^{0*x^130},$
 $g^{8*x^192} + g^{7*x^160} + g^{4*x^144} + g^{2*x^136} + g^{0*x^132},$
 $g^{7*x^192} + g^{6*x^160} + g^{3*x^144} + g^{1*x^136} + g^{0*x^132},$
 $g^{6*x^192} + g^{5*x^160} + g^{2*x^144} + g^{0*x^136},$
 $g^{5*x^192} + g^{4*x^160} + g^{1*x^144} + g^{0*x^136},$
 $g^{4*x^192} + g^{3*x^160} + g^{0*x^144},$
 $g^{3*x^192} + g^{2*x^160} + g^{0*x^144},$
 $g^{2*x^192} + g^{1*x^160} + g^{0*x^144},$
 $g^{1*x^192} + g^{0*x^160} + g^{0*x^144},$
 $g^{0*x^192} + g^{0*x^160} + g^{0*x^144},$
 $g^{1*x^192} + g^{1*x^160} + g^{1*x^144} + g^{1*x^136} + g^{1*x^132} + g^{1*x^130} + g^{1*x^128} + g^{1*x^126} + g^{1*x^124} + g^{1*x^122} + g^{1*x^120} + g^{1*x^118} + g^{1*x^116} + g^{1*x^114} + g^{1*x^112} + g^{1*x^110} + g^{1*x^108} + g^{1*x^106} + g^{1*x^104} + g^{1*x^102} + g^{1*x^100} + g^{1*x^98} + g^{1*x^96} + g^{1*x^94} + g^{1*x^92} + g^{1*x^90} + g^{1*x^88} + g^{1*x^86} + g^{1*x^84} + g^{1*x^82} + g^{1*x^80} + g^{1*x^78} + g^{1*x^76} + g^{1*x^74} + g^{1*x^72} + g^{1*x^70} + g^{1*x^68} + g^{1*x^66} + g^{1*x^64} + g^{1*x^62} + g^{1*x^60} + g^{1*x^58} + g^{1*x^56} + g^{1*x^54} + g^{1*x^52} + g^{1*x^50} + g^{1*x^48} + g^{1*x^46} + g^{1*x^44} + g^{1*x^42} + g^{1*x^40} + g^{1*x^38} + g^{1*x^36} + g^{1*x^34} + g^{1*x^32} + g^{1*x^30} + g^{1*x^28} + g^{1*x^26} + g^{1*x^24} + g^{1*x^22} + g^{1*x^20} + g^{1*x^18} + g^{1*x^16} + g^{1*x^14} + g^{1*x^12} + g^{1*x^10} + g^{1*x^8} + g^{1*x^6} + g^{1*x^4} + g^{1*x^2} + g^{1*x^0},$
 $g^{1*x^192} + g^{0*x^160} + g^{0*x^144} + g^{0*x^136} + g^{0*x^132} + g^{0*x^130} + g^{0*x^128} + g^{0*x^126} + g^{0*x^124} + g^{0*x^122} + g^{0*x^120} + g^{0*x^118} + g^{0*x^116} + g^{0*x^114} + g^{0*x^112} + g^{0*x^110} + g^{0*x^108} + g^{0*x^106} + g^{0*x^104} + g^{0*x^102} + g^{0*x^100} + g^{0*x^98} + g^{0*x^96} + g^{0*x^94} + g^{0*x^92} + g^{0*x^90} + g^{0*x^88} + g^{0*x^86} + g^{0*x^84} + g^{0*x^82} + g^{0*x^80} + g^{0*x^78} + g^{0*x^76} + g^{0*x^74} + g^{0*x^72} + g^{0*x^70} + g^{0*x^68} + g^{0*x^66} + g^{0*x^64} + g^{0*x^62} + g^{0*x^60} + g^{0*x^58} + g^{0*x^56} + g^{0*x^54} + g^{0*x^52} + g^{0*x^50} + g^{0*x^48} + g^{0*x^46} + g^{0*x^44} + g^{0*x^42} + g^{0*x^40} + g^{0*x^38} + g^{0*x^36} + g^{0*x^34} + g^{0*x^32} + g^{0*x^30} + g^{0*x^28} + g^{0*x^26} + g^{0*x^24} + g^{0*x^22} + g^{0*x^20} + g^{0*x^18} + g^{0*x^16} + g^{0*x^14} + g^{0*x^12} + g^{0*x^10} + g^{0*x^8} + g^{0*x^6} + g^{0*x^4} + g^{0*x^2} + g^{0*x^0},$

$g^{*213*x^*192 + g^*92*x^*160 + g^*40*x^*144 + g^*186*x^*136 + g^*76*x^*132 + g^*179*x^*130 + g^*200*x^*129 + g^*181*x^*80 + g^*212*x^*66 + g^*94*x^*48 + g^*83*x^*36 + g^*27*x^*34 + g^*183*x^*20 + g^*231*x^*12 + g^*126*x^*9 + g^*253*x^*10 + g^*172*x^*9 + g^*220*x^*6 + g^*29*x^*3, g^{*192*x^*192 + g^*40*x^*160 + g^*18*x^*144 + g^*202*x^*136 + g^*246*x^*132 + g^*48*x^*130 + g^*129*x^*129 + g^*127*x^*96 + g^*159*x^*80 + g^*213*x^*72 + g^*73*x^*68 + g^*191*x^*66 + g^*18*x^*36 + g^*131*x^*34 + g^*78*x^*33 + g^*38*x^*20 + g^*173*x^*18 + g^*161*x^*12 + g^*225*x^*9 + g^*64*x^*6 + g^*215*x^*3, g^{*183*x^*192 + g^*156*x^*160 + g^*221*x^*144 + g^*223*x^*136 + g^*191*x^*130 + g^*149*x^*129 + g^*109*x^*96 + g^*107*x^*72 + g^*153*x^*68 + g^*42*x^*66 + g^*198*x^*65 + g^*71*x^*36 + g^*203*x^*40 + g^*167*x^*36 + g^*153*x^*34 + g^*71*x^*33 + g^*206*x^*24 + g^*184*x^*18 + g^*136*x^*10 + g^*51*x^*9 + g^*192*x^*5 + g^*160*x^*3, g^{*126*x^*192 + g^*206*x^*160 + g^*114*x^*144 + g^*165*x^*136 + g^*14*x^*132 + g^*140*x^*130 + g^*73*x^*129 + g^*75*x^*68 + g^*140*x^*66 + g^*125*x^*132 + g^*237*x^*130 + g^*79*x^*129 + g^*12*x^*96 + g^*120*x^*80 + g^*14*x^*72 + g^*242*x^*68 + g^*197*x^*40 + g^*134*x^*36 + g^*204*x^*24 + g^*213*x^*30 + g^*42*x^*24 + g^*179*x^*20 + g^*20*x^*18 + g^*253*x^*18 + g^*14*x^*17 + g^*113*x^*12 + g^*113*x^*5 + g^*224*x^*3, g^{*115*x^*192 + g^*202*x^*160 + g^*194*x^*144 + g^*145*x^*136 + g^*12*x^*132 + g^*179*x^*130 + g^*17*x^*96 + g^*146*x^*80 + g^*223*x^*72 + g^*6*x^*68 + g^*23*x^*12 + g^*179*x^*130 + g^*17*x^*96 + g^*146*x^*80 + g^*210*x^*20 + g^*197*x^*18 + g^*219*x^*17 + g^*102*x^*12 + g^*255*x^*10 + g^*126*x^*5 + g^*29*x^*3, g^{*68*x^*192 + g^*85*x^*160 + g^*181*x^*136 + g^*98*x^*132 + g^*120*x^*130 + g^*178*x^*129 + g^*85*x^*68 + g^*147*x^*72 + g^*39*x^*18 + g^*105*x^*68 + g^*152*x^*34 + g^*81*x^*36 + g^*161*x^*33 + g^*89*x^*17 + g^*172*x^*12 + g^*12*x^*9 + g^*38*x^*6 + g^*15*x^*5 + g^*29*x^*3, g^{*223*x^*192 + g^*238*x^*160 + g^*235*x^*144 + g^*182*x^*136 + g^*19*x^*129 + g^*120*x^*96 + g^*42*x^*72 + g^*178*x^*68 + g^*94*x^*65 + g^*114*x^*48 + g^*222*x^*36 + g^*120*x^*18 + g^*15*x^*36 + g^*19*x^*18 + g^*206*x^*24 + g^*167*x^*36 + g^*117*x^*10 + g^*51*x^*9 + g^*192*x^*5 + g^*160*x^*3, g^{*226*x^*192 + g^*68*x^*160 + g^*20*x^*144 + g^*17*x^*136 + g^*113*x^*130 + g^*16*x^*129 + g^*124*x^*96 + g^*109*x^*80 + g^*11*x^*72 + g^*163*x^*68 + g^*12*x^*48 + g^*209*x^*36 + g^*10*x^*20 + g^*19*x^*18 + g^*16*x^*12 + g^*11*x^*8 + g^*11*x^*6 + g^*11*x^*5 + g^*23*x^*3, g^{*12*x^*192 + g^*223*x^*160 + g^*111*x^*136 + g^*10*x^*129 + g^*19*x^*130 + g^*11*x^*129 + g^*11*x^*96 + g^*11*x^*72 + g^*11*x^*68 + g^*11*x^*66 + g^*11*x^*64 + g^*11*x^*62 + g^*11*x^*60 + g^*11*x^*58 + g^*11*x^*56 + g^*11*x^*54 + g^*11*x^*52 + g^*11*x^*50 + g^*11*x^*48 + g^*11*x^*46 + g^*11*x^*44 + g^*11*x^*42 + g^*11*x^*40 + g^*11*x^*38 + g^*11*x^*36 + g^*11*x^*34 + g^*11*x^*32 + g^*11*x^*30 + g^*11*x^*28 + g^*11*x^*26 + g^*11*x^*24 + g^*11*x^*22 + g^*11*x^*20 + g^*11*x^*18 + g^*11*x^*16 + g^*11*x^*14 + g^*11*x^*12 + g^*11*x^*10 + g^*11*x^*8 + g^*11*x^*6 + g^*11*x^*4 + g^*11*x^*2 + g^*11*x^*0 + g^*11*x^*1 + g^*11*x^*3 + g^*11*x^*5 + g^*11*x^*7 + g^*11*x^*9 + g^*11*x^*11 + g^*11*x^*13 + g^*11*x^*15 + g^*11*x^*17 + g^*11*x^*19 + g^*11*x^*21 + g^*11*x^*23 + g^*11*x^*25 + g^*11*x^*27 + g^*11*x^*29 + g^*11*x^*31 + g^*11*x^*33 + g^*11*x^*35 + g^*11*x^*37 + g^*11*x^*39 + g^*11*x^*41 + g^*11*x^*43 + g^*11*x^*45 + g^*11*x^*47 + g^*11*x^*49 + g^*11*x^*51 + g^*11*x^*53 + g^*11*x^*55 + g^*11*x^*57 + g^*11*x^*59 + g^*11*x^*61 + g^*11*x^*63 + g^*11*x^*65 + g^*11*x^*67 + g^*11*x^*69 + g^*11*x^*71 + g^*11*x^*73 + g^*11*x^*75 + g^*11*x^*77 + g^*11*x^*79 + g^*11*x^*81 + g^*11*x^*83 + g^*11*x^*85 + g^*11*x^*87 + g^*11*x^*89 + g^*11*x^*91 + g^*11*x^*93 + g^*11*x^*95 + g^*11*x^*97 + g^*11*x^*99 + g^*11*x^*101 + g^*11*x^*103 + g^*11*x^*105 + g^*11*x^*107 + g^*11*x^*109 + g^*11*x^*111 + g^*11*x^*113 + g^*11*x^*115 + g^*11*x^*117 + g^*11*x^*119 + g^*11*x^*121 + g^*11*x^*123 + g^*11*x^*125 + g^*11*x^*127 + g^*11*x^*129 + g^*11*x^*131 + g^*11*x^*133 + g^*11*x^*135 + g^*11*x^*137 + g^*11*x^*139 + g^*11*x^*141 + g^*11*x^*143 + g^*11*x^*145 + g^*11*x^*147 + g^*11*x^*149 + g^*11*x^*151 + g^*11*x^*153 + g^*11*x^*155 + g^*11*x^*157 + g^*11*x^*159 + g^*11*x^*161 + g^*11*x^*163 + g^*11*x^*165 + g^*11*x^*167 + g^*11*x^*169 + g^*11*x^*171 + g^*11*x^*173 + g^*11*x^*175 + g^*11*x^*177 + g^*11*x^*179 + g^*11*x^*181 + g^*11*x^*183 + g^*11*x^*185 + g^*11*x^*187 + g^*11*x^*189 + g^*11*x^*191 + g^*11*x^*193 + g^*11*x^*195 + g^*11*x^*197 + g^*11*x^*199 + g^*11*x^*201 + g^*11*x^*203 + g^*11*x^*205 + g^*11*x^*207 + g^*11*x^*209 + g^*11*x^*211 + g^*11*x^*213 + g^*11*x^*215 + g^*11*x^*217 + g^*11*x^*219 + g^*11*x^*221 + g^*11*x^*223 + g^*11*x^*225 + g^*11*x^*227 + g^*11*x^*229 + g^*11*x^*231 + g^*11*x^*233 + g^*11*x^*235 + g^*11*x^*237 + g^*11*x^*239 + g^*11*x^*241 + g^*11*x^*243 + g^*11*x^*245 + g^*11*x^*247 + g^*11*x^*249 + g^*11*x^*251 + g^*11*x^*253 + g^*11*x^*255 + g^*11*x^*257 + g^*11*x^*259 + g^*11*x^*261 + g^*11*x^*263 + g^*11*x^*265 + g^*11*x^*267 + g^*11*x^*269 + g^*11*x^*271 + g^*11*x^*273 + g^*11*x^*275 + g^*11*x^*277 + g^*11*x^*279 + g^*11*x^*281 + g^*11*x^*283 + g^*11*x^*285 + g^*11*x^*287 + g^*11*x^*289 + g^*11*x^*291 + g^*11*x^*293 + g^*11*x^*295 + g^*11*x^*297 + g^*11*x^*299 + g^*11*x^*301 + g^*11*x^*303 + g^*11*x^*305 + g^*11*x^*307 + g^*11*x^*309 + g^*11*x^*311 + g^*11*x^*313 + g^*11*x^*315 + g^*11*x^*317 + g^*11*x^*319 + g^*11*x^*321 + g^*11*x^*323 + g^*11*x^*325 + g^*11*x^*327 + g^*11*x^*329 + g^*11*x^*331 + g^*11*x^*333 + g^*11*x^*335 + g^*11*x^*337 + g^*11*x^*339 + g^*11*x^*341 + g^*11*x^*343 + g^*11*x^*345 + g^*11*x^*347 + g^*11*x^*349 + g^*11*x^*351 + g^*11*x^*353 + g^*11*x^*355 + g^*11*x^*357 + g^*11*x^*359 + g^*11*x^*361 + g^*11*x^*363 + g^*11*x^*365 + g^*11*x^*367 + g^*11*x^*369 + g^*11*x^*371 + g^*11*x^*373 + g^*11*x^*375 + g^*11*x^*377 + g^*11*x^*379 + g^*11*x^*381 + g^*11*x^*383 + g^*11*x^*385 + g^*11*x^*387 + g^*11*x^*389 + g^*11*x^*391 + g^*11*x^*393 + g^*11*x^*395 + g^*11*x^*397 + g^*11*x^*399 + g^*11*x^*401 + g^*11*x^*403 + g^*11*x^*405 + g^*11*x^*407 + g^*11*x^*409 + g^*11*x^*411 + g^*11*x^*413 + g^*11*x^*415 + g^*11*x^*417 + g^*11*x^*419 + g^*11*x^*421 + g^*11*x^*423 + g^*11*x^*425 + g^*11*x^*427 + g^*11*x^*429 + g^*11*x^*431 + g^*11*x^*433 + g^*11*x^*435 + g^*11*x^*437 + g^*11*x^*439 + g^*11*x^*441 + g^*11*x^*443 + g^*11*x^*445 + g^*11*x^*447 + g^*11*x^*449 + g^*11*x^*451 + g^*11*x^*453 + g^*11*x^*455 + g^*11*x^*457 + g^*11*x^*459 + g^*11*x^*461 + g^*11*x^*463 + g^*11*x^*465 + g^*11*x^*467 + g^*11*x^*469 + g^*11*x^*471 + g^*11*x^*473 + g^*11*x^*475 + g^*11*x^*477 + g^*11*x^*479 + g^*11*x^*481 + g^*11*x^*483 + g^*11*x^*485 + g^*11*x^*487 + g^*11*x^*489 + g^*11*x^*491 + g^*11*x^*493 + g^*11*x^*495 + g^*11*x^*497 + g^*11*x^*499 + g^*11*x^*501 + g^*11*x^*503 + g^*11*x^*505 + g^*11*x^*507 + g^*11*x^*509 + g^*11*x^*511 + g^*11*x^*513 + g^*11*x^*515 + g^*11*x^*517 + g^*11*x^*519 + g^*11*x^*521 + g^*11*x^*523 + g^*11*x^*525 + g^*11*x^*527 + g^*11*x^*529 + g^*11*x^*531 + g^*11*x^*533 + g^*11*x^*535 + g^*11*x^*537 + g^*11*x^*539 + g^*11*x^*541 + g^*11*x^*543 + g^*11*x^*545 + g^*11*x^*547 + g^*11*x^*549 + g^*11*x^*551 + g^*11*x^*553 + g^*11*x^*555 + g^*11*x^*557 + g^*11*x^*559 + g^*11*x^*561 + g^*11*x^*563 + g^*11*x^*565 + g^*11*x^*567 + g^*11*x^*569 + g^*11*x^*571 + g^*11*x^*573 + g^*11*x^*575 + g^*11*x^*577 + g^*11*x^*579 + g^*11*x^*581 + g^*11*x^*583 + g^*11*x^*585 + g^*11*x^*587 + g^*11*x^*589 + g^*11*x^*591 + g^*11*x^*593 + g^*11*x^*595 + g^*11*x^*597 + g^*11*x^*599 + g^*11*x^*601 + g^*11*x^*603 + g^*11*x^*605 + g^*11*x^*607 + g^*11*x^*609 + g^*11*x^*611 + g^*11*x^*613 + g^*11*x^*615 + g^*11*x^*617 + g^*11*x^*619 + g^*11*x^*621 + g^*11*x^*623 + g^*11*x^*625 + g^*11*x^*627 + g^*11*x^*629 + g^*11*x^*631 + g^*11*x^*633 + g^*11*x^*635 + g^*11*x^*637 + g^*11*x^*639 + g^*11*x^*641 + g^*11*x^*643 + g^*11*x^*645 + g^*11*x^*647 + g^*11*x^*649 + g^*11*x^*651 + g^*11*x^*653 + g^*11*x^*655 + g^*11*x^*657 + g^*11*x^*659 + g^*11*x^*661 + g^*11*x^*663 + g^*11*x^*665 + g^*11*x^*667 + g^*11*x^*669 + g^*11*x^*671 + g^*11*x^*673 + g^*11*x^*675 + g^*11*x^*677 + g^*11*x^*679 + g^*11*x^*681 + g^*11*x^*683 + g^*11*x^*685 + g^*11*x^*687 + g^*11*x^*689 + g^*11*x^*691 + g^*11*x^*693 + g^*11*x^*695 + g^*11*x^*697 + g^*11*x^*699 + g^*11*x^*701 + g^*11*x^*703 + g^*11*x^*705 + g^*11*x^*707 + g^*11*x^*709 + g^*11*x^*711 + g^*11*x^*713 + g^*11*x^*715 + g^*11*x^*717 + g^*11*x^*719 + g^*11*x^*721 + g^*11*x^*723 + g^*11*x^*725 + g^*11*x^*727 + g^*11*x^*729 + g^*11*x^*731 + g^*11*x^*733 + g^*11*x^*735 + g^*11*x^*737 + g^*11*x^*739 + g^*11*x^*741 + g^*11*x^*743 + g^*11*x^*745 + g^*11*x^*747 + g^*11*x^*749 + g^*11*x^*751 + g^*11*x^*753 + g^*11*x^*755 + g^*11*x^*757 + g^*11*x^*759 + g^*11*x^*761 + g^*11*x^*763 + g^*11*x^*765 + g^*11*x^*767 + g^*11*x^*769 + g^*11*x^*771 + g^*11*x^*773 + g^*11*x^*775 + g^*11*x^*777 + g^*11*x^*779 + g^*11*x^*781 + g^*11*x^*783 + g^*11*x^*785 + g^*11*x^*787 + g^*11*x^*789 + g^*11*x^*791 + g^*11*x^*793 + g^*11*x^*795 + g^*11*x^*797 + g^*11*x^*799 + g^*11*x^*801 + g^*11*x^*803 + g^*11*x^*805 + g^*11*x^*807 + g^*11*x^*809 + g^*11*x^*811 + g^*11*x^*813 + g^*11*x^*815 + g^*11*x^*817 + g^*11*x^*819 + g^*11*x^*821 + g^*11*x^*823 + g^*11*x^*825 + g^*11*x^*827 + g^*11*x^*829 + g^*11*x^*831 + g^*11*x^*833 + g^*11*x^*835 + g^*11*x^*837 + g^*11*x^*839 + g^*11*x^*841 + g^*11*x^*843 + g^*11*x^*845 + g^*11*x^*847 + g^*11*x^*849 + g^*11*x^*851 + g^*11*x^*853 + g^*11*x^*855 + g^*11*x^*857 + g^*11*x^*859 + g^*11*x^*861 + g^*11*x^*863 + g^*11*x^*865 + g^*11*x^*867 + g^*11*x^*869 + g^*11*x^*871 + g^*11*x^*873 + g^*11*x^*875 + g^*11*x^*877 + g^*11*x^*879 + g^*11*x^*881 + g^*11*x^*883 + g^*11*x^*885 + g^*11*x^*887 + g^*11*x^*889 + g^*11*x^*891 + g^*11*x^*893 + g^*11*x^*895 + g^*11*x^*897 + g^*11*x^*899 + g^*11*x^*901 + g^*11*x^*903 + g^*11*x^*905 + g^*11*x^$

g²⁷*x¹⁹² + g²⁵⁴*x¹⁶⁰ + g²¹⁷*x¹⁴⁴ + g²²³*x¹³⁶ + g⁴⁵*x¹³² + g⁸²*x¹³⁰ + g¹³⁰*x¹²⁹ + g¹²⁴*x⁹⁶ + g²⁵⁴*x⁸⁰ + g²¹⁹*x⁷² + g²⁰*x⁶⁸ + g¹⁵³*x⁶⁶ + g³⁹*x⁴⁸ + g²³*x³⁴ + g⁶¹*x²⁴ + g¹⁸⁹*x²⁰ + g⁷⁰*x¹⁷ + g⁴*x¹² + g¹²⁰*x⁹ + g⁵⁶*x⁶ + g²²⁷*x⁵ + g¹²*x³, g⁷⁶*x¹⁹² + g²⁰⁴*x¹⁶⁰ + g²⁰⁷*x¹⁴⁴ + g¹⁶⁹*x¹³⁶ + g¹⁴⁴*x¹³⁰ + g⁵²*x¹²⁹ + g⁷⁷*x⁹⁶ + g¹⁹⁵*x⁸⁰ + g⁵*x⁷² + g⁴⁰*x⁶⁸ + g¹³⁹*x⁶⁶ + g¹²⁵*x⁶⁵ + g¹⁷⁸*x⁴⁸ + g⁷²*x⁴⁰ + g²⁴⁰*x³⁴ + g¹⁹³*x³³ + g⁸⁷*x²⁴ + g¹⁹²*x¹⁸ + g²⁰⁵*x¹² + g⁷⁷*x¹⁰ + g⁵⁵*x⁹ + g⁷⁰*x⁶ + g⁶¹*x⁵ + g³⁶*x³, g⁸³*x¹⁹² + g¹²⁸*x¹⁶⁰ + g¹⁸⁴*x¹³⁶ + g²²⁵*x¹³² + g¹⁷⁴*x¹³⁰ + g¹⁵⁸*x⁹⁶ + g¹⁰⁷*x⁸⁰ + g⁶⁸*x⁶⁶ + g⁶²*x⁴⁸ + g²¹⁴*x³⁴ + g¹⁰⁸*x³³ + g¹⁵¹*x²⁴ + g¹⁸⁷*x¹⁸ + g²⁰⁹*x¹⁰ + g²⁹*x⁹ + g²⁰⁴*x⁶ + g⁷⁶*x⁵ + g²²⁹*x³, g¹⁰⁵*x¹⁹² + g¹⁷⁰*x¹⁶⁰ + g¹⁷⁴*x¹³⁶ + g²⁵²*x¹³² + g⁵⁷*x¹³⁰ + g²²⁷*x¹²⁹ + g⁹⁵*x⁹⁶ + g¹⁷⁵*x⁸⁰ + g²¹⁴*x⁷² + g⁶⁸*x⁶⁶ + g⁸⁰*x⁶⁵ + g⁷⁴*x⁴⁸ + g⁷*x³⁶ + g¹¹⁶*x³⁴ + g¹⁹⁹*x³³ + g²⁴²*x²⁴ + g¹¹⁴*x²⁰ + g¹⁹⁵*x¹⁸ + g¹²⁹*x¹⁷ + g¹⁰²*x¹² + g⁷*x¹⁰ + g²⁸*x⁹ + g²¹⁷*x⁶ + g²⁵⁰*x¹⁸ + g⁹⁷*x¹⁷ + g³¹*x¹² + g¹⁹⁶*x¹⁰ + g³⁷*x⁹ + g¹⁵⁹*x⁶ + g¹⁸⁹*x⁵ + g¹⁹¹*x³, g²²¹*x¹⁹² + g¹⁴¹*x¹⁶⁰ + g²²*x¹³⁶ + g¹¹¹*x¹³⁰ + g²¹⁹*x⁹⁶ + g¹²⁰*x⁷² + g⁸⁴*x⁶⁸ + g¹⁴⁰*x⁴⁰ + g¹⁸⁹*x³⁶ + g²³*x²⁴ + g²²³*x²⁰ + g⁵⁶*x¹² + g¹⁰⁴*x¹⁰ + g⁶¹*x⁹ + g¹⁴*x⁶ + g¹²*x³, g¹⁶¹*x¹⁹² + g¹⁸³*x¹⁶⁰ + g²¹³*x¹³⁶ + g¹²*x¹³² + g²⁶*x¹³⁰ + g⁷⁷*x⁹⁶ + g²⁰⁵*x⁶⁶ + g⁴³*x⁴⁸ + g¹⁷*x³⁶ + g⁵⁸*x³⁴ + g¹⁸*x²⁴ + g²⁵*x¹² + g²*x¹⁰ + g²⁰*x⁹ + g⁶²*x⁶ + g²²*x⁵, g⁵¹*x¹⁹² + g¹²*x¹⁶⁰ + g²⁵⁰*x¹³⁶ + g⁵⁰*x¹³⁰ + g¹¹⁹*x⁹⁶ + g¹⁴⁴*x⁷² + g⁹⁷*x⁴⁰ + g²⁴²*x³⁴ + g¹⁵⁰*x³³ + g²³*x²⁴ + g¹⁶⁰*x²⁰ + g⁴⁷*x¹⁸ + g¹³⁵*x¹² + g⁹²*x¹⁰ + g¹⁴⁵*x⁶ + g¹⁰⁹*x³, g²²⁹*x¹⁹² + g²⁴⁸*x¹⁶⁰ + g¹⁷⁹*x¹³² + g⁷⁰*x¹³⁰ + g²⁰²*x¹²⁹ + g¹²⁶*x⁹⁶ + g¹¹⁹*x⁸⁰ + g⁴⁵*x⁶⁶ + g¹⁸⁷*x⁵³ + g²³⁸*x³³ + g²⁰*x¹⁷ + g¹⁸⁹*x¹² + g²²⁴*x¹⁰ + g⁸⁵*x⁹ + g¹²³*x⁶ + g²²⁷*x⁵, g²²³*x¹⁹² + g²⁴*x¹⁶⁰ + g⁷⁴*x¹³⁶ + g¹⁹³*x¹³⁰ + g¹⁰⁷*x⁹⁶ + g²²⁸*x⁸⁰ + g³⁴*x³⁶ + g³⁷*x²⁴ + g⁹¹*x²⁰ + g¹⁹⁹*x¹⁸ + g¹⁹*x¹⁶ + g²⁵³*x⁵ + g²³¹*x³, g⁸⁰*x¹⁹² + g⁹⁹*x¹⁶⁰ + g¹⁶⁹*x¹³⁶ + g²⁴³*x¹³⁰ + g⁹³*x⁹⁶ + g¹⁸⁸*x⁸⁰ + g¹⁷⁴*x⁶⁸ + g⁶¹*x⁴⁰ + g²⁴⁸*x³⁶ + g¹⁰⁸*x³³ + g²⁰*x²⁴ + g¹⁸*x²⁰ + g¹⁷*x¹⁸ + g¹¹⁶*x¹² + g²⁴⁶*x¹⁰ + g⁶⁶*x⁹ + g²¹¹*x⁸ + g¹⁴¹*x¹⁹² + g²⁵*x¹⁶⁰ + g⁷⁶*x¹³⁶ + g²¹¹*x¹³⁰ + g¹⁰⁵*x⁹⁶ + g²⁴⁹*x⁸⁰ + g¹⁰*x⁶⁶ + g¹⁴⁵*x⁵³ + g²⁵¹*x³³ + g²³*x²⁴ + g⁹*x²⁰ + g¹⁹³*x¹⁸ + g¹⁹*x¹⁶ + g²⁵⁰*x⁵ + g²²³*x³, g¹⁸⁵*x¹⁹² + g¹⁸⁹*x¹⁶⁰ + g²³*x¹³⁶ + g¹⁹¹*x¹³⁰ + g¹⁰*x⁹⁶ + g¹⁷⁸*x⁸⁰ + g¹⁵⁵*x⁶⁸ + g¹⁸⁷*x⁵³ + g²¹*x²⁴ + g¹⁷*x²⁰ + g¹⁸*x¹⁸ + g¹¹*x¹² + g²⁴⁶*x¹⁰ + g⁶³*x⁹ + g²⁰*x⁸ + g¹⁷*x⁶ + g¹⁴*x³, g⁶⁴*x¹⁹² + g⁵²*x¹⁶⁰ + g¹³⁰*x¹⁴⁴ + g¹⁴*x¹³⁰ + g²⁰⁵*x⁸⁰ + g³¹*x¹² + g¹⁶⁴*x⁶⁸ + g⁵²*x⁶⁵ + g¹⁰⁶*x⁴⁸ + g¹⁴⁵*x⁴⁰ + g¹⁸*x³³ + g¹⁹¹*x³⁰ + g²²*x¹⁸ + g⁴*x¹⁶ + g²⁶*x¹³ + g¹⁶*x¹⁰ + g¹⁸⁴*x⁹ + g²²*x⁶ + g¹⁷*x³, g⁵¹*x¹⁹² + g¹²*x¹⁶⁰ + g²⁵⁰*x¹³⁶ + g⁵⁰*x¹³⁰ + g¹¹⁹*x⁹⁶ + g¹⁴⁴*x⁷² + g⁹⁷*x⁴⁰ + g²⁴²*x³⁴ + g¹⁵⁰*x³³ + g²³*x²⁴ + g¹⁶⁰*x²⁰ + g⁴⁷*x¹⁸ + g¹³⁵*x¹² + g⁹²*x¹⁰ + g¹⁴⁵*x⁶ + g¹⁰⁹*x³, g²³⁹*x¹⁹² + g²⁴⁸*x¹⁶⁰ + g¹⁷⁹*x¹³² + g⁷⁰*x¹³⁰ + g²⁰²*x¹²⁹ + g¹²⁶*x⁹⁶ + g¹¹⁹*x⁸⁰ + g⁴⁵*x⁶⁶ + g¹⁸⁷*x⁵³ + g²³⁸*x³³ + g²⁰*x¹⁷ + g¹⁸⁹*x¹² + g²²⁴*x¹⁰ + g⁸⁵*x⁹ + g¹²³*x⁶ + g²²⁷*x⁵, g²²³*x¹⁹² + g²⁴*x¹⁶⁰ + g⁷⁴*x¹³⁶ + g¹⁹³*x¹³⁰ + g¹⁰⁷*x⁹⁶ + g²²⁸*x⁸⁰ + g³⁴*x³⁶ + g³⁷*x²⁴ + g⁹¹*x²⁰ + g¹⁹⁹*x¹⁸ + g¹⁹*x¹⁶ + g²⁵³*x⁵ + g²³¹*x³, g⁸⁰*x¹⁹² + g⁹⁹*x¹⁶⁰ + g¹⁶⁹*x¹³⁶ + g²⁴³*x¹³⁰ + g⁹³*x⁹⁶ + g¹⁸⁸*x⁸⁰ + g¹⁷⁴*x⁶⁸ + g⁶¹*x⁴⁰ + g²⁴⁸*x³⁶ + g¹⁰⁸*x³³ + g²⁰*x²⁴ + g¹⁸*x²⁰ + g¹⁷*x¹⁸ + g¹¹⁶*x¹² + g²⁴⁶*x¹⁰ + g⁶⁶*x⁹ + g²¹*x⁸ + g¹⁴¹*x¹⁹² + g²⁵*x¹⁶⁰ + g⁷⁶*x¹³⁶ + g²¹¹*x¹³⁰ + g¹⁰⁵*x⁹⁶ + g²⁴⁹*x⁸⁰ + g¹⁰*x⁶⁶ + g¹⁴⁵*x⁵³ + g²⁵¹*x³³ + g²³*x²⁴ + g⁹*x²⁰ + g¹⁹³*x¹⁸ + g¹⁹*x¹⁶ + g²⁵⁰*x⁵ + g²²³*x³, g¹⁸⁵*x¹⁹² + g¹⁸⁹*x¹⁶⁰ + g²³*x¹³⁶ + g¹⁹¹*x¹³⁰ + g¹⁰*x⁹⁶ + g¹⁷⁸*x⁸⁰ + g¹⁵⁵*x⁶⁸ + g¹⁸⁷*x⁵³ + g²¹*x²⁴ + g¹⁷*x²⁰ + g¹⁸*x¹⁸ + g¹¹*x¹² + g²⁴⁶*x¹⁰ + g⁶³*x⁹ + g²⁰*x⁸ + g¹⁷*x⁶ + g¹⁴*x³, g⁶⁴*x¹⁹² + g⁵²*x¹⁶⁰ + g¹³⁰*x¹⁴⁴ + g¹⁴*x¹³⁰ + g²⁰⁵*x⁸⁰ + g³¹*x¹² + g¹⁶⁴*x⁶⁸ + g⁵²*x⁶⁵ + g¹⁰⁶*x⁴⁸ + g¹⁴⁵*x⁴⁰ + g¹⁸*x³³ + g¹⁹¹*x³⁰ + g²²*x¹⁸ + g⁴*x¹⁶ + g²⁶*x¹³ + g¹⁶*x¹⁰ + g¹⁸⁴*x⁹ + g²²*x⁶ + g¹⁷*x³, g⁵¹*x¹⁹² + g¹²*x¹⁶⁰ + g²⁵⁰*x¹³⁶ + g⁵⁰*x¹³⁰ + g¹¹⁹*x⁹⁶ + g¹⁴⁴*x⁷² + g⁹⁷*x⁴⁰ + g²⁴²*x³⁴ + g¹⁵⁰*x³³ + g²³*x²⁴ + g¹⁶⁰*x²⁰ + g⁴⁷*x¹⁸ + g¹³⁵*x¹² + g⁹²*x¹⁰ + g¹⁴⁵*x⁶ + g¹⁰⁹*x³, g²³⁹*x¹⁹² + g²⁴⁸*x¹⁶⁰ + g¹⁷⁹*x¹³² + g⁷⁰*x¹³⁰ + g²⁰²*x¹²⁹ + g¹²⁶*x⁹⁶ + g¹¹⁹*x⁸⁰ + g⁴⁵*x⁶⁶ + g¹⁸⁷*x⁵³ + g²³⁸*x³³ + g²⁰*x¹⁷ + g¹⁸⁹*x¹² + g²²⁴*x¹⁰ + g⁸⁵*x⁹ + g¹²³*x⁶ + g²²⁷*x⁵, g²²³*x¹⁹² + g²⁴*x¹⁶⁰ + g⁷⁴*x¹³⁶ + g¹⁹³*x¹³⁰ + g¹⁰⁷*x⁹⁶ + g²²⁸*x⁸⁰ + g³⁴*x³⁶ + g³⁷*x²⁴ + g⁹¹*x²⁰ + g¹⁹⁹*x¹⁸ + g¹⁹*x¹⁶ + g²⁵³*x⁵ + g²³¹*x³, g⁸⁰*x¹⁹² + g⁹⁹*x¹⁶⁰ + g¹⁶⁹*x¹³⁶ + g²⁴³*x

