applied to digraphs in the usual 26-letter alphabet. The enciphering matrix was determined using the Diffie–Hellman key exchange method, as follows. Working in the prime field of 3602561 elements, your correspondent sent you $g^b = 983776$. Your randomly chosen Diffie-Hellman exponent a is 1082389. Finally, you agree to get a matrix from a key number $K_E \in \mathbf{F}_{3602561}$ by writing the least nonnegative residue of K_E modulo 26^4 in the form $a \cdot 26^3 + b \cdot 26^2 + c \cdot 26 + d$ (where a, b, c, d are digits in the base 26). If the resulting matrix is not invertible modulo 26, replace K_E by $K_E + 1$ and try again. Take as the enciphering matrix the first invertible matrix that arises from the successive integers starting with K_E.

(a) Use this information to find the enciphering matrix.

(b) Find the deciphering matrix, and read the message.

5. Suppose that each user A has a secret pair of transformations f_A and f_A^{-1} from $\mathcal{P}$ to $\mathcal{P}$, where $\mathcal{P}$ is a fixed set of plaintext message units. They want to transmit information securely using the Massey–Omura technique, i.e., Alice sends $f_A(P)$ to Bob, who then sends $f_B(f_A(P))$ back to her, and so on. Give the conditions that the system of f_A's must satisfy in order for this to work.

6. Let p be the Fermat prime 65537, and let $g = 5$. You receive the message (29095, 23846), which your friend composed using the ElGamal cryptosystem in $\mathbf{F}_p^*$, using your public key g^a. Your secret key, needed for deciphering, is $a = 13908$. You have agreed to convert integers in $\mathbf{F}_p$ to trigraphs in the 31-letter alphabet of Exercise 3 by writing them to the base 31, the digits in the 31^2-, the $31-$ and $1-$ place being the numerical equivalents of the three letters in the trigraph. Decipher the message.

7. (a) Show that choosing $\mathbf{F}_p$ with $p = 2^{2^k} + 1$ a Fermat prime is an astoundingly bad idea, by constructing a polynomial time algorithm for solving the discrete log problem in $\mathbf{F}_p^*$ (i.e., an algorithm which is polynomial in $log\, p$). To do this, suppose that g is a generator (e.g., 5 or 7, as shown in Exercise 15 of § II.2) and for a given a you want to find x, where $0 \le x < p - 1 = 2^{2^k}$, such that $g^x \equiv a \bmod p$. Write x in binary, and pattern your algorithm after the algorithm for extracting square roots modulo p that was described at the end of § II.2.

   (b) Find a big-O estimate (in terms of p) for the number of bit operations required to find the integer x by means of the algorithm in part (a).

   (c) Use the algorithm in part (a) to find the value of k in Exercise 6.

8. Suppose that your plaintext message units are 18-letter blocks written in the usual 26-letter alphabet, where the numerical equivalent of such a block is an 18-digit base-26 integer (written in order of decreasing powers of 26). You receive the message

$$(82746592004375034872957717,\ 164063768437915425954819351),$$