Similarly, the "random" number e prime to $\varphi(n)$ can be chosen by first generating a random (odd) integer with an appropriate number of bits, and then successively incrementing it until one finds an e with $g.c.d.(e, \varphi(n)) = 1$. (Alternately, one can perform primality tests until one finds a prime e, say between $max(p, q)$ and $\varphi(n)$; such a prime must necessarily satisfy $g.c.d.(e, \varphi(n)) = 1$.)

Thus, each user A chooses two primes p_A and q_A and a random number e_A which has no common factor with $(p_A - 1)(q_A - 1)$. Next, A computes $n_A = p_A q_A$, $\varphi(n_A) = n_A + 1 - p_A - q_A$, and also the multiplicative inverse of e_A modulo $\varphi(n_A)$: $d_A \underset{\text{def}}{=} e_A^{-1} \ mod\ \varphi(n_A)$. She makes public the enciphering key $K_{E,A} = (n_A,\ e_A)$ and conceals the deciphering key $K_{D,A} = (n_A,\ d_A)$. The enciphering transformation is the map from $\mathbf{Z}/n_A\mathbf{Z}$ to itself given by $f(P) \equiv P^{e_A} \ mod\ n_A$. The deciphering transformation is the map from $\mathbf{Z}/n_A\mathbf{Z}$ to itself given by $f^{-1}(C) \equiv C^{d_A} \ mod\ n_A$. It is not hard to see that these two maps are inverse to one another, because of our choice of d_A. Namely, performing f followed by f^{-1} or f^{-1} followed by f means raising to the $d_A e_A$-th power. But, because $d_A e_A$ leaves a remainder of 1 when divided by $\varphi(n_A)$, this is the same as raising to the 1-st power (see the corollary of Proposition I.3.5, which gives this in the case when P has no common factor with n_A; if $g.c.d.(P, n_A) > 1$, see Exercise 6 below).

From the description in the last paragraph, it seems that we are working with sets $\mathcal{P} = \mathcal{C}$ of plaintext and ciphertext message units that vary from one user to another. In practice, we would probably want to choose $\mathcal{P}$ and $\mathcal{C}$ uniformly throughout the system. For example, suppose we are working in an N-letter alphabet. Then let $k < \ell$ be suitably chosen positive integers, such that, for example, N^k and N^ℓ have approximately 200 decimal digits. We take as our plaintext message units all blocks of k letters, which we regard as k-digit base-N integers, i.e., we assign them numerical equivalents between 0 and N^k. We similarly take ciphertext message units to be blocks of ℓ letters in our N-letter alphabet. Then each user must choose his/her large primes p_A and q_A so that $n_A = p_A q_A$ satisfies $N^k < n_A < N^\ell$. Then any plaintext message unit, i.e., integer less than N^k, corresponds to an element in $\mathbf{Z}/n_A\mathbf{Z}$ (for any user's n_A); and, since $n_A < N^\ell$, the image $f(P) \in \mathbf{Z}/n_A\mathbf{Z}$ can be uniquely written as an ℓ-letter block. (Not all ℓ-letter blocks can arise — only those corresponding to integers less than n_A for the particular user's n_A.)

Example 1. For the benefit of a reader who doesn't have a computer handy (or does not have good multiple precision software), we shall sacrifice realism and choose most of our examples so as to involve relatively small integers. Choose $N = 26$, $k = 3$, $\ell = 4$. That is, the plaintext consists of trigraphs and the ciphertext consists of four-graphs in the usual 26-letter alphabet. To send the message "YES" to a user A with enciphering key $(n_A, e_A) = (46927, 39423)$, we first find the numerical equivalent of "YES," namely: $24 \cdot 26^2 + 4 \cdot 26 + 18 = 16346$, and then compute $16346^{39423} \ mod\ 46927$, which is $21166 = 1 \cdot 26^3 + 5 \cdot 26^2 + 8 \cdot 26 + 2 =$ "BFIC."