the RSAEncryption class is a simple implementation of the RSA encryption scheme with key size 1024 bits. The class first generates two distinct 512-bit probable primes p and q, then calculates the product of these two primes (n = pq), and calculates the Euler totient function (phi = (p-1)(q-1)).

Next, the class uses an encryption exponent e = 65537 and checks if it is relatively prime to phi(n). If it is not, it goes back to the first step and generates new values for p and q. Once a suitable value for e is found, the class computes the value for the decryption exponent d, which is the multiplicative inverse of e mod phi(n).

The encrypt() and decrypt() methods are used to perform encryption and decryption, respectively. The main() method demonstrates how these methods can be used to encrypt and decrypt a meaningful message with a minimum of 20 words.

To compile and run the source code:

-Compilation

javac RSAGUI.java KA.java

-Running

java RSAGUI