RSA:-

RSA is a first public cryptography algorithm. Its basic idea is factorizing a large number will be very difficult. This algorithm consists of two key private key and public key.

Public key:-

As every one knows about it. Let a text denoted as “C” can be encrypted by the formula

**Encrypted = ( C ^ e ) mod n**

‘n’ can be calculated as p\*q;

As p and q are large prime number.

1 < e < phi(n) \*\*Note e should not be a factor of ‘n’

Phi(n) = (p-1) \* (q-1)

**Decrypted = (Encrypted ^ d ) mod n**

\*\*\*d x e mod n=1

‘d’ can be found by extended euclidean algorithm

Phi(n)x + ey = gcd ( phi(n), e )

**Finding ‘d’**

Let D=phi(n)

Let A0=1,B0=0,A1=0,B1=1;

While (D!=1){

Kk =(int)(Dk-1/Dk)

Ak+1 = Ak-1 - (Ak \* Kk)

Bk+1 = Bk-1 - (Bk \* Kk)

Dk+1 = Dk-1 - (Dk \* Kk)

}

The ‘B’ value where D=1 is the ‘d’ value

If d > phi(n)

d = d mod phi(n)

If d<0

d= d+ phi(n);

Else

d will remain same.

**Algorithm :**

1. Read character by character at a time like “hello world”

We consider the ascii value (ascii value-32+1)

h - 72 is as ‘C’ and apply **Encrypted = ( C ^ e ) mod n**

if ‘H’ 08 padding will be done to make 2 digit.

For the variable length generation of encryption padding will be used for uniformity

1. Then write the encrypted thing in the file and read next character
2. Reverse process will be used for decryption

**Decrypted = (Encrypted ^ d ) mod n**

**Pro type of function:**

Void function(filename){

Read character to char and implement.

}

Implementation using BigInteger class in java

Source:

<https://www.youtube.com/watch?v=plxMklEvlCU>

<https://www.geeksforgeeks.org/rsa-algorithm-cryptography/>

<https://en.wikipedia.org/wiki/RSA_(cryptosystem)>