R.S.A. Public Key Encryption

The RSA algorithm was developed in 1978 by Ron ***R***ivest, Adi ***S***hamir, and Leonard ***A***dleman from whose names the algorithm name is derived. It was designed to replace the less secure encryption method used by the National Bureau of Standards.

RSA is a public key cryptosystem in which the numeric key(s) are published for all to see. In a public key cryptosystem, the numeric keys are used to encrypt the message, and the receiver of the encrypted message is the only one who knows the secret decryption key.

Suppose Alice wants to send an encrypted message with plaintext m to Bob. Here is how we would use the RSA system to do this

1. Bob will first choose two secret prime numbers p and q (usually hundreds of digits long) and compute $n = pq$ and $\phi(n) = (p-1)(q-1)$.
2. Bob then chooses an *encryption exponent* e such that $\gcd(e, \phi) = 1$
3. Bob then computes the *decryption exponent* d such that $de \equiv 1 \ (mod\ \phi)$.
4. Bob makes n and e public, and keeps p, q, d secret.
5. Alice encryptes her plaintext message m as $c \equiv m^e (mod\ n)$ and sends c to Bob.
6. Bob decrypts by computing $m \equiv c^d (mod\ n)$.

Let's give this a try using a TI-Nspire program. Below is a copy of the program we will use. You can also enter a program like this in your TI-84. It uses the simple "TI-Basic" language.

RSA Program for TI-Nspire or TI-8x

```
Define rsa(a,b)=
Prgm
:a→p:b→q
:p*q→n
:Disp "N=p*q=",n
:(p-1)*(q-1)→phi
:Disp "phi=",phi
:Lbl exponent
:randInt(2,phi-1)→e
:If gcd(e,n)≠1
:Goto exponent
:0→d
:2→test
:While d=0 and test≤phi
:If mod(e*test,phi)=1 Then
:test→d
:Else
:test+1→test
:If test=phi Then
:Goto exponent
:EndIf
:EndIf
:EndWhile
:Disp "Encryption Exponent: e=",e
:Disp "Decryption Exponent: d=",d
:EndPrgm
```