COSC413 Programming Assignment on Cryptosystems

Problem: Implement the RSA public-key cryptosystem in single precision and multiple precision, and do some experiments.

The problem is broken down into the following sub-problems

Setting up the system:

(1) Set up two large prime numbers p and q by the probabilistic method. Then compute \( n = pq \). (p : 50 digits, q: 50 digits, n : 100 digits for multiple precision)

(2) Determine e and d using the Euclidean algorithm. (e, n) and (d, n) are encryption and decryption keys.

Encryption and Decryption:

(1) An English message \( L \) is encoded into a numerical message \( M \) by your method.

(2) Cryptogram \( C = M^e \mod n \) is computed.

........ ( transmission )

(3) The integer message \( M = C^d \mod n \) is recovered.

(4) The above \( M \) is decoded into the original English message \( L \).

By setting up two sets of keys do experiment on secure and authentic message transmission. An English message of A4 size must be used.

Note: The procedures must have an open architecture for accommodating single integer arithmetic and multiple precision arithmetic. You should start from the single precision version to see if the RSA scheme works OK. Sample programs for multiple precision arithmetic and primality test are attached, which are not most efficient and should be taken as just a starting point.

Present in hard copy the following

(1) Outline of the RSA system
(2) Documentation of your software for the RSA system, including a user’s manual.
(3) Experimental results with measurements of time.
(4) Source code

Present the following source program and English text file electronically.

(A) single.c and text.txt. After “gcc single”, the sessions should go by hitting the return key.
(B) multiple.c and text.txt. The same note as above

Due dates: Single Precision, 5:00 pm, 4 April 2008, worth 15%. Present (1), (2), (4) and (A)
Multiple Precision, 5:00 pm, 26 April 2008, worth 15%. Present (2), (3), (4) and (B)