CS3235 Tutorial for week 5 (Sept 10-Sept 14, 2007)

September 6, 2007

Present your answer (on paper) to the following tutorial question at the beginning of the tutorial session.

2 (To be handed in). An essential component of the RSA cryptographic scheme is raising a large number \( x \) to a large power \( y \) (modulo some other number \( n \)). We could do this by just multiplying \( x \) by itself \( y-1 \) times (we will call this method A), but this is not fast. Find a faster method (method B) for calculating \( x^y \).

(a) Estimate the time complexity of method A and method B using big O notation.
(b) Given that a multiplication takes 1mS, and assuming that all other operations are instantaneous, estimate the time to calculate \( x^y \) using each method, where \( y \) is about \( 2^{300} \).

Please come to the tutorial ready to present your answers to these questions as well:

2 (Do not hand in). Using your knowledge of entropy, Calculate the entropy in bits/symbol of a source transmitting 32 different printable characters, with the probabilities of E, T, A, O, N, S, H, R being \( \frac{1}{5} \), \( \frac{1}{5} \), \( \frac{1}{10} \), \( \frac{1}{10} \), \( \frac{1}{10} \), \( \frac{1}{10} \), \( \frac{1}{10} \), and \( \frac{1}{10} \) respectively and the other 24 characters being evenly distributed. Estimate the average size of a 32 character message if you could use the best encoding scheme.

3 (Do not hand in). Use Fermat’s theorem to show the inverse of \( q = 153 \) in the integers modulo 103387. Describe exactly how you did this. What limits your technique?

4 (Do not hand in). The following sentence is encrypted using a monoalphabetic substitution cipher. What this means is that an ordinary English sentence has been encrypted by changing each letter to some other letter.

(a) Decrypt the sentence:

```
BJSB NZCC SEKI D TBGTCDHI LZVB HG HBI LEWVH VBHJAIYH NBG IFDZCV HBZV HG BZF
```

(b) Describe the technique you used to decrypt it.

You may find that the decrypt tool at https://www-appn.comp.nus.edu.sg/~cs3235/mono.cgi may be useful. It measures the frequency of letters in the sentence. (You figure out how to use it)