a revised

Assignment One (Due date 17/4/2020: 12:00 pm) (25 marks)

Objective: To implement **Affine Cipher and S-DES** in any programming language with your preference. **But** you are required to finish this assignment ***independently*** and strictly follow the ***requirements***. In your assignment, please pay attention to your writing skills in grammar checking, typos, etc.

1. Implement the revised **Affine Cipher** (change mod 26 to mod 27) with one programing language and make sure that the following requirements are satisfied. **(8 marks)**

   - You need to have a **separate code file** to testify whether a given key (a, b) is eligible.
   - You should have separate functions for encryption and decryption and also your code should work for both encryption and decryption for any given eligible key. (After decryption, make sure that the original plaintext is recovered correctly).
   - You code should be capable to encrypt and decrypt both **capital and lower case letters** (ignore other non-letter symbols).
   - You code should work for the given test file. (Able to encrypt and decrypt correctly).

   After implementing your code, *you MUST answer the following questions in your e-copy submission.*

   - Compute all possible eligible keys you can use and justify your computation.
   - If your code works properly with some selected eligible key, show recovered plaintext after decryption and compare with the original plaintext file; (the test file is given in the unit website). State the possible reasons if your code is not working properly.
   - **Mathematically prove** the decrypted message equals to the original message with critical logical reasoning.
   - Use your code in the Tutorial 1, print out the letter distribution with a graph chart for the given test file.
   - Explain briefly how your code can skip/ignore non-letter symbols.
   - Submit your code with detailed code comments.

- Print out the first page and last page of decrypted file in your e-copy and compare them with the original plaintext, are they the same?

2. Implement **DES** in any programming language. The requirements are follows: **(12 Marks)**
    - Your code should be able to encrypt and decrypt all possible characters on a keyboard.
    - The key for encryption and decryption is required to be any combination of characters in a keyboard with finite length (You need to do padding or chopping if necessary.)
    - You are required to implement **key generation**, **switch function** and **F(Sub-key, R)** as three **separate** functions (You should have these three functions separately in your hard copy) and then combine all of these operations to achieve **DES.**
    - Make sure your code works properly for the given test file.
    - Change the elements in the two **S-boxes** with the required constraint (S-Box elements are required to be between 0-3) and check whether your code still works as expected.

    Now you can answer the following questions in your e-copy submission after you have done all above.

    - Mathematically **prove DES** works (After decryption, you can obtain the original plaintext).
    - State/depict your pseudo code structure based on the three separate required functions.
    - State **clearly** what is the main difficulty in the process of your programming if your code is **not working properly**?
    - Print out and submit the outputs for encryption and decryption for a test file given in the unit website. (**Only the first page and last pages are required in hard copy**.)
    - Print out and submit one e-copy of your code **with structure explanation by using the three functions** (make sure this structure looks nice).
    - Try to do encryption and decryption with a key of all 0's, and report your findings.

You need to report your design/progress and demonstrate your code in the lab for above two questions. Make sure that answers in your e-copy are consistent with your live demo in lab.

3. Based on your understanding on the lecture notes 1-4 (from lecture one to lecture four), answer the following questions. **(5 Marks)**
    - Describe what kind of possible threats you can overcome with DES and justify your reply.
    - Explain what you have done for source coding in Question 2 for DES.
    - If you want to achieve the highest probability of error-correction in information transmission, tell us what you should do when you design a channel encoder.