# CPE-321
# Introduction of Computer Security
## *Warming up Python with Historical Crypto*

**Objectives**

The objectives for this lab assignment are as follows:

- To crack classical ciphers (Caesar, Vigenère, and mono-alphabetic substitution cipher);

**Programming Tools:** Python

**Requirements**

Please read the requirements carefully, and finish the lab assignment accordingly:

- Implement these classical ciphers using Phyton. Along with the built-in Python functions you may only use the math and string libraries. • Do not use any other libraries, online tools, cracking programs, or code you did not write.
- In the tarred file **encrypted.tar.gz** you will find 9 files. Each of these files has been encrypted with a historic cipher. You need to decrypt them.

**Questions**

1. What are the advantage and disadvantage of these classical cipher? 2. Will you consider using these classical ciphers for your content protection? Why or why not?

3. Discuss your experience during the crack implementations.

**In Your Report**

Please address the following in your report:

- Describe Caesar, Vigenère, and mono-alphabetic substitution cipher;
- Describe how to implement the encryption and save the output files;
- Describe the crack implementation:
    - File name;
    - Key and which cipher;
        - For Caesar the encryption key should be a number between 0 and 25 (assume A=0, B=1, C=2 and so on);
        - For Vigerene the encryption key should be a word
        - For the mono alphabetic the key should be a 26 character string
    - Process used to decrypt the file – this should be a simple description of the process you used to crack the cipher, referencing any code you wrote;
    - Code you wrote to decrypt the file (you must use at least some code for each task)
- Show the demo: screen shot of the result and code
- Answer the questions and attach your source code

**Useful algorithms:** You may want to consider implementing the following algorithms to help with decrypting the files

1. A Caesar Cipher decryption algorithm;
2. A Vigerene decryption algorithm;
3. A mono-alphabetic substitution decryption algorithm;
4. Letter frequency counter;
5. Chi-squared test;
6. Index of coincidence;
7. Digraph frequency count;

8. Repeat analysis to determine Vigerene key length.

**Helpful hints:**

1. Some files have space removed;
2. The input texts for files include English texts;
3. At least one input text includes scientific names;
4. Vigerene ciphers have key lengths of 5, 9, and 13;
5. At least one is a proper noun;
6. At least one is a sentence without spaces;
7. Some English text does not follow normal frequency distribution

**Submission:** Include a report with the information for each file you decrypted. Make sure that you describe your cracking process in enough detail to demonstrate you understand how to crack the cipher. Also, make sure a classmate could reproduce your results from your process description and code.