Problem Set 1

This problem set is due **at 8:00pm** on **Friday, September 20, 2024**.

- The TAs will provide a detailed document describing how you should submit your PDF and code on Google Classroom (we may use gradescope for some things). Make sure you read it! We suggest that you perform a trial submission prior to the deadline to make sure that everything works for you – you can overwrite that submission with a new one up to the deadline.
- We require that written solutions are submitted as a PDF file, **typeset on LaTeX**, using the template available on Google Classroom. You must **show your work** for written solutions. Each solution should start on a new page.
- We will occasionally ask you to "give an algorithm" to solve a problem. Your write-up should take the form of a short essay. Start by defining the problem you are solving and stating what your results are. Then provide: (a) a description of the algorithm in English and, if helpful, pseudo-code; (b) a proof sketch for the correctness of the algorithm; and (c) an analysis of the running time.
- We will give full credit **only** for correct solutions that are described clearly and convincingly.

Problem 1-1. Caesar Cipher [5 points]

Caesar wants to arrange a secret meeting with Antony, either at the Tiber (the river) or at the Colosseum (the arena). He sends the ciphertext EVIRE. However, Antony does not know the key, so he tries all possibilities. Where will he meet Caesar?

Problem 1-2. Affine Cipher [10 points]

The ciphertext UCR was encrypted using the affine function $9x + 2 \mod 26$. Find the plaintext (show your work).

Problem 1-3. Transposition Ciphers [30 points]

(a) [10 points] The following ciphertext was encrypted using the *columnar transposition encryption* with the key **"CRYPTOGRAPHY"**. Find the plaintext (Show your work).

`ypdrphitysitvosndesuepgennooyeibhamielibgonlitsoa`

(b) [20 points] Write **C** code for columnar transposition encryption and decryption (key and key-size can be choose randomly or by user input).

Problem 1-4. Hill Cipher [20 points]

(a) [15 points] Eve captures Bob's Hill cipher machine, which uses a 2-by-2 matrix $M \mod 26$. She tries a chosen plaintext attack and finds that the plaintext BA encrypts to HC and the plaintext ZZ encrypts to GT. What is the matrix M? (Assuming $A = 0, B = 1$, etc.)

(b) [5 points] Can a $K = \begin{bmatrix} 7 & 2 \\ 1 & 4 \end{bmatrix}$ be used for encryption? Justify your answer.

Problem 1-5. Perfect Security [30 points]

Alice wants to send a message to Bob. Alice has recently taken a cryptography class, and is much enamoured with the concept of perfect security. She does her duty and reads the lecture slides given out in class - with special attention to the example on perfect security. Unfortunately she needs to send five messages, not three, namely - $\{5, 6, 7, 8, 9\}$ are the messages she wishes to send.

(a) Alice decides to use the Caesar Cipher, with the same three keys $\{0, 1, 2\}$ as given in the example in Lecture Slides 3-1. Her messages have the probabilities $\{1/3, 1/3, 1/6, 1/12, 1/12\}$. Can Alice find an encryption method under the given conditions that will allow her to reach perfect security? Why or why not?

(b) Alice learns a new language. In this language, all her messages are the same as before, but their probabilities are equal. Is it possible to either modify (if necessary) the previous system (if you found one), or to create one (if you couldn't find one), in such a way that the newer cryptosystem is perfectly secure?

(c) If it was possible to create such a system - prove that the system you found for Alice is perfectly secure.
If it was not possible to create such a system - find a way to modify Alice's system to give it perfect security.
What do these exercises tell you about perfect security? Can you draw some larger conclusions about this type of security from these exercises?

Problem 1-6. Multiple Encipherment [15 points]

Beff Jezos, the founder of Ganga, is sending a message to Melon Usk using one of the following cryptosystems. In fact, Beff is bored and his plaintext consists of the letter a repeated a few hundred times.

Zark Muckerberg (the owner of BookFace), who is spying on them, knows what system is being used, but not the key, and intercepts the ciphertext.

For systems (a), (b), and (c), state how Zark will recognize that the plaintext is one repeated letter and decide whether or not Zark can deduce the letter and the key. (Note: For system (c), the solution very much depends on the fact that the repeated letter is a, rather than $b, c, \ldots$)

(a) Shift

(b) Affine

(c) Hill (2×2)

Problem 1-7. Shift Cipher [20 points]

Decrypt the following ciphertexts, which was encrypted using a simple shift cipher:

(a) `uryczrvzgenccrqvafvqrnpnrfnepvcurenaqpnagtrgbhg`

(b) `pfldljksvjgvvufwczxyksvtrljvkzdvjkfgjnyvezcffbrkpflyrggpmrcvekzevjurp`

(c) `hgnodxnthmsdqbdossghrrdbqdssqzmrlhrrhnmvhsgntszmxdqqnqsghrsqzmrlhrrhnmgzrsqzudkkdczlhkkhnmkhfgsxdzqrsnhmenqlxntsgzsvdzqdbnlhmfrnnm`

(d) `khzzaxwuwjeowckkzkjaatyalpbkniahkjiahkjiahkjiahkjiahkj`

Problem 1-8. Vigenere Cipher [30 points]

The following pieces of text were encrypted using the Vigenere method, using key lengths of at most 6. Write and submit code to decrypt these ciphertexts. Note: The code should be well commented and supplied with a readme on how to run it.

(a)
```
qivjukosqegnyiytxypshzewjsnsdpeybsuiranshzewjsnsdvusdvozqhasg
hexhvtdrynjyirlrrnfpekjbsuhucnjyirlrrnfveylrsdgbinjyirlrrnfwi
lqbsuqlisfqhhzuxytxaewhroxwvasjirxwsltyiytxontzxhjuyljvenivsd
tlectpqiypinylwwmdxirosoplrgkrvytxaoswkeywlixivordrytwlewjyyn
mysyzensdxeqocozkswnpjejomnlzensdqaphcozxrdjuwtfqhnjyirlrrnfj
mvjbsuzsreahvgtqraqhxytxhobq
```

(b)
```
text file has been provided
```

(c)
```
hdsfgvmkoowafweetcmfthskucaqbilgjofmaqlgspvatvxqbiryscpcfrmvsw
rvnqlszdmgaoqsakmlupsqforvtwvdfcjzvgsoaoqsacjkbrsevbelvbksarls
cdcaarmnvrysywxqgvellcyluwwveoafgclazowafojdlhssfiksepsoywxafo
wlbfcsocylngqsyzxgjbmlvgrggokgfgmhlmejabsjvgmlnrvqzcrggcrghgeu
pcyfgtydycjkhqluhgxgzovqswpdvbwsffsenbxapasgazmyuhgsfhmftayjxm
wznrsofrsoaopgauaaarmftqsmahvqecev
```

Problem 1-9. Enigma Plugboard [10 points]

The Enigma machine played a major role in secure communication in the World War era. If used properly, the time required to break the Enigma encryption would be some orders of magnitude beyond the ability to check by hand, as it was done by the allies. There are a number of mechanisms in the Enigma which made it secure, the most basic one is what's referred to as the Plugboard. Plugboard is the name of a simple setup which has two sets of the 26 alphabets, any letter can be connected to another letter in the other set using a plug. The output then has these two letters exchanged. For example, if R and U were plugged then *rural* would come out as *urual*. Therefore, the plugboard would be used to pair letters in two. Given k plugs, where $k < 13$, how many unique plugboard settings exist using these plugs? Provide proof if necessary.