CSIT 6000Q- Blockchain and Smart Contracts

Assigment #1
(Due: 11:59pm on Sun Day, Oct. 15, 2023)

September 22, 2023

1 Problem 1

Decrypt the following ciphers by simple substitution. Please use the following application:
`https://cryptii.com/pipes/caesar-cipher`

1.1 Problem 1(A) 5 points

Nudq sgd ozrs edv xdzqr, xnt gzud bnmrhrsdmskx gdzqc sgd sdql ‘aknbjbgzhm sdbgmnknfx,’ oqnazakx qdfzqchmf bqxosnbtqqdmbhdr khjd Ahsbnhm. Hm ezbs, xnt lzx ad zrjhmf xntqrdke, “Vgzs hr aknbjbgzhm sdbgmnknfx?” Hs rddlr khjd aknbjbgzhm hr z okzshstcd, ats hm z gxonsgdshbzk rdmrd, zr sgdqd hr mn qdzk ldzmhmf sgzs sgd kzxlzm bzm tmcdqrszmc dzrhkx. Hs hr hlodqzshud sn zmrvdq “vgzs hr aknbjbgzhm sdbgmnknfx, “hmbktchmf sgd sdbgmnknfx sgzs hr trdc, gnv hs vnqjr, zmc gnv hs’r adbnlhmf uhszk hm sgd chfhszk vnqkc. Zr aknbjbgzhm bnmshmtdr sn fqnv zmc adbnld lnqd trdq-eqhdmckx, sgd nmtr hr nm xnt sn kdzqm sghr dunkuhmf sdbgmnknfx sn oqdozqd enq sgd etstqd. He xnt zqd mdv sn aknbjbgzhm, sgdm sghr hr sgd qhfgs okzsenql sn fzhm rnkhc entmczshnmzk jmnvkdcfd. Hm sghr zqshbkd, xnt kdzqm gnv sn zmrvdq sgd ptdrshnm, “Vgzs hr aknbjbgzhm sdbgmnknfx?” Xnt’kk zkrn kdzqm gnv aknbjbgzhm vnqjr, vgx hs’r hlonqszms, zmc gnv xnt bzm trd sghr ehdkc sn zcuzmbd xntq bzqddq.

1.2 Problem 1(B) 7 points

hjeedhtndjpgtigpchutggxcvbdctnidndjgupbxandgugxtcshugdbndjgqpczprrdjcindjldjasadv
xciddcaxctqpczxcvpcsigpchutgiwtpbdjciidiwtdiwtgetghdcjhxcviwtxgprrdjcicjbqtglwt
ciwtigpchprixdcxhsdctndjgqpczjespithiwtigpchprixdcgtrdgshxihttbhhxbeattcdjvwgxv
wiiwtgtxhpeditcixpaxhhjtlwxrwbdhidujhctvatriiwthtinethduigpchprixdchrpcqtipbetg
tslxiwktgnfjxrzanetdeatlwdpgtupbxaxpglxiwiwxhigjiwpgtduitclpgndujhxcviwthtineth
duigpchprixdchwtcrtiwttkdajixdcduiwxgsepginepnbtcipeeaxrpixdchxcgtrtcintpghqjii
wxhkjactgpqxaxinxhthhtcixpaanlwnqadrzrwpxcitrwcdadvnlphrgtpitsitrwcdadvxrpaanqa
drzrwpxcxhpsxvxipaatsvtgiwpixhvpxcxcvpadidupiitcixdcpcsigprixdcgtrtcianqjilwnwp
hxiqtrdbthdedejapgltaaatihsxvxcidxiidupiwdbiwtlwdatrdcrteigtrdgszttexcvduspippc
sigpchprixdchpgtprgjrxpaepgiduiwtqjhxcthhduitciwxhxcudgbpixdcxhwpcsatsxcwdjhtdg
ephhtsiwgdjvwpiwxgsepginaxztqgdztghqpcztghdgaplntghxcrgtphxcvixbtrdhidgqdiwdciw
tqjhxcthhudgijcpitanqadrzrwpxcpkdxshiwxhadcvegdrthhpcsuprxaxipithiwtuphitgbdktb

tciduiwtigpchprixdciwtgtqnhpkxcvqdiwixbtpcsbdctnbdhietdeatphhjbtqadrzrwpxcpcsqx
irdxcrpcqtjhtsxcitgrwpcvtpqanqjixcgtpaxiniwpihcdiiwtrphtqadrzrwpxcxhiwtitrwcdad
vnrpepqatduhjeedgixcvkpgxdjhpeeaxrpixdchgtapitsidbjaixeatxcsjhigxthaxztuxcpcrth
jeeanrwpxcbpcjuprijgxcvtirqjiqxirdxcxhprjggtcrniwpigtaxthdcqadrzrwpxcitrwcdadvnidqthtrjgt

1.3 Problem 1(C) 8 points

zgxbwozixpgsmgakwvaqabwnbewsmgazqdibmsmgivlcjtqksmgpmamsmgapmtxqvxmznwzuq
voackkmaanctbzivaikbqwvajmbemmvbewxizbqmaikpqvlqdqlcitpiabpmambewsmgaepqkpbpmg
cambwxzwlckmiamkczmlqoqbitqlmvbqbgzmnmzmvkmpqaamkczmlqlmvbqbgqabpmuwabqu
xwzbivbiaxmkbwntwkskpiqvbmkpvwtwogvbpmewztlwnkzgxbwkczzmvkgbpqaqlmvbqbgqazm
nmzzmlbwialqoqbitaqovibczmivlqacamlnwzicbpwzqhqvoivlkwvbzwttqvobzivaikbqwvapmlqoq
bitaqovibczmqaumzomleqbpbpmxmmzbwxmmzvmbewzsitizomvcujmzwnqvlqdqlcitaepwikbiai
cbpwzqbqmacambpmlqoqbitaqovibczmqvwzlmzbwzmikpikwvamvacawvbzivaikbqwvaiuwvow
bpmzqaacmapmvbpmgicbpwzqhmilmitqbqakmzbqnqmljgiuibpmuibqkitdmzqnqkibqwvepqkpz
mactbaqviackkmaanctamkczmlbzivaikbqwvjmbemmvbpmbewvmbewzskwvvmkbmlxizbqmaw
bwacuqbcxtwkskpiqvcamzamuxtwgkzgxbwozixpgsmgabwxmznwzulqnnmzmvbbgxmawnlqoq
bitqvbmzikbqwvawdmzbpmxmmzbwxmmzvmbewzs

2 Problem 2 20 points

The following ciphertext is obtained using a simple substitution cipher (punctuations and blanks are deleted), and the language is English. Decrypt it by using the frequency distribution of English letters. You may also use the fact that certain diagrams (e.g., an, en, er, es, he) appear more frequently. This may help you. Please also give some details about how you decrypt it.

DIJOFTFSFNBJOEFSUIFPSFNJTPOFPGUIF KFXFMTPGNBUIFNBUJDTJUJTBQFSGFDU DPNCJOBUJPOPGCFBVUZBOEVUJMJUZUIJT CPPLUFMMTBCPVUUIFDIJOFTFSFNBJOEFS UIFPSFNJUTCBDLHSPVOEBOEQIJMPTPQIZ IJTUPSZHFOFSBMJABUJPOTBOENPTU JNQPSUBOUMZJUTBQQMJDBUJPOT

Please use the applications at the following URLs for computing the frequencies of single letters and diagrams:

- `https://www.101computing.net/frequency-analysis/`
- `https://www.braingle.com/brainteasers/codes/frequencyanalysis.php`

3 Problem 3 25 points

We identify each English letter with an integer between 0 and 25 as follows:

A	B	..	Y	Z
0	2	..	24	25

Take any pair (k_0, k_1) of integers such that $gcd(k_0, 26)$ = 1 and $0 \leq k_i \leq 25$, and define the 1-to-1 mapping f by

$$f(x) = (x \times k_0 + k_1) \, mod \, 26.$$

So f is a substitution function of the English alphabet.

A simple substitution cipher based on f is a 5-tuple $(\mathcal{M}, \mathcal{C}, \mathcal{K}, E_k, D_k)$, where $\mathcal{M}$ and $\mathcal{C}$ are the set of all finite strings of English letters; $\mathcal{K}$ is the set of all possible f:

- $k = (k_0, k_1) \in \mathcal{K}$ is the encryption and decryption key,
- For a message $m = m_0 m_1 m_2 ...$,
- $E_k(m)$ = $f(m_0)f(m_1)f(m_2)...$, For a ciphertext $c = c_0 c_1 c_2 ...$,
- $D_k(c)$ = $f^{-1}(c_0)f^{-1}(c_1)f^{-1}(c_2)...$

Example : Use the secret key $(k_0, k_1) = (3, 1)$ to encrypt the message “missyou” results ciphertext ”LZDDVRJ”.

Design encryption and decryption algorithm using pseudo code.

4 Problem 4 25 points

How do certain simple constructions fail to ensure the cryptographic strength of the proposed hashing algorithms?

More precisely, the collisions for the proposed methods are easily found. Let the input data be of the form X = $(X_0, X_1, X_2, ..., X_{n-1})$ where each X_i is a byte. Consider the following hash function :

$$h(X) = X_0 + X_1 + X_2 + ... + X_{n-1}$$

, where $+$ stands for bitwise modulo two addition. Is this a secure hashing method in the sense that collisions are hard to find?

5 Problem 5 10 points

Write a use-case scenario where a smart contract addresses a real-world problem.