MATH 75: Cryptography **Spring 2022**

PSET 3 — April 22, 2022

Prof. Asher Auel ***Student: Amittai Siavava***

Credit Statement

I worked on these problems alone, with reference to class notes and the following books:

(a) ***The Code Book*** by **Simon Singh**.

(b) ***Cryptography*** by **Simon Rubinsen-Salzedo**

Problems

1. Consider the affine cipher with $\mathcal{P} = \mathcal{C} = \mathbb{Z}/n\mathbb{Z}$.

(a) Suppose $n = 541$ and we take the key $(a, b) = (34, 71)$. Encrypt the plaintext $m = 204$, and decrypt the ciphertext $c = 431$.

The encryption of $m = 204$ is 515

$$\begin{aligned} c &= a \cdot p + b \\ &= 34 \cdot 204 + 71 \\ &= 7007 \\ &\equiv 515 \pmod{541} \end{aligned}$$

The decryption of $c = 431$ is 297

$$\begin{aligned} c &\equiv a \cdot p + b \pmod{n} \\ 431 &\equiv 34p + 71 \pmod{541} \\ 360 &\equiv 34p \pmod{541} \\ p &= \frac{360 + 541k}{34} \qquad \mid p, k \in \mathbb{Z}^+ \end{aligned}$$

We need to find a value for k such that $34 | (541k + 360)$. This is easily calculated to be $k = 18$.

$$\begin{aligned} p &= \frac{360 + 541 \cdot 18}{34} \\ p &= 297 \end{aligned}$$

(b) Eve intercepts a ciphertext from Alice and through espionage she learns that the letter $x \in \mathcal{P}$ is encrypted as $y \in \mathcal{C}$ in this message. Show that Eve can decrypt the message using $O(n)$ trials.

Suppose Eve knows that a letter $x \in \mathcal{P}$ is encrypted as $y \in \mathcal{C}$ in the message.
Then, Eve knows that $a \cdot x + b \pmod{n} \equiv y \pmod{n}$ for some $a, b \in \mathbb{Z}/n\mathbb{Z}$, where (a, b) are the keys of the Affine Cipher.

$$\begin{aligned} ax + b &\equiv y \pmod{n} \\ ax + b &\equiv y \\ ax &\equiv y - b \end{aligned}$$

Eve can safely assume that $0 \leq b \leq n - 1$ (since adding any number $x \geq n$ is equivalent to adding $x \mod n$). Eve can therefore iterate through all the n possible values of b and test the matching value for a. She is guaranteed to find the real key.

(c) Now suppose that (contrary to Kerckhoffs's principle) the integer n is not public knowledge. Is the affine cipher still vulnerable if Eve manages to steal a plaintext/ciphertext pair? How might Eve break the system?

Without knowing n, the problem becomes much harder to break. However, if Eve knows at least 3 different plaintext/ciphertext pairs, she can use them to guess a value for n.
Say, for instance, p_1, p_2, p_3 are the plaintexts and c_1, c_2, c_3 are the ciphertexts:

$$\begin{aligned} c_1 &\equiv a \cdot p_1 + b \pmod{n} \Rightarrow a \cdot p_1 + b - c_1 \equiv 0 \pmod{n} \\ c_2 &\equiv a \cdot p_2 + b \pmod{n} \Rightarrow a \cdot p_2 + b - c_2 \equiv 0 \pmod{n} \\ c_3 &\equiv a \cdot p_3 + b \pmod{n} \Rightarrow a \cdot p_3 + b - c_3 \equiv 0 \pmod{n} \end{aligned}$$

$$\begin{bmatrix} c_1 & p_1 & 1 \\ c_2 & p_2 & 1 \\ c_3 & p_3 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -a \\ -b \end{bmatrix} \equiv \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \pmod{n}$$

$$\begin{vmatrix} c_1 & p_1 & 1 \\ c_2 & p_2 & 1 \\ c_3 & p_3 & 1 \end{vmatrix} \equiv 0 \pmod{n}$$

$$c_1(p_2 - p_3) - c_2(p_1 - p_3) + c_3(p_1 - p_2) \equiv 0 \pmod{n}$$

Since $\{c_1, c_2, c_3\}$ and the corresponding plaintexts $\{p_1, p_2, p_3\}$ are all known, Eve can find a value congruent to 0 in $\mathbb{Z}/n\mathbb{Z}$ and use it to find n, after which she can easily crack the encryption.

2.

Encrypt the message

Why is a raven like a writing desk

using the Vigenère cipher with keyword `rabbithole`.

The encryption is ‘‘NHZJATYOGIELJLMTDFTXZNHEMLR’’

Algorithm

I wrote a program to encrypt and decrypt per the Vigenère cipher.

```
-- | Get the "vigenere complement" of a character.
--
-- The complement of ’A’ is itself (shift by 0),
--
-- the complement of ’B’ is ’Z’ (shift by 1 and -1), etc.
invChar :: Char -> Char
invChar char = chr (ord ’Z’ - (charToInt char - 1))
```

For convenience, we can define a function that maps invChar over a word:

```
-- | Get the "vigenere complement" of a word.
--
-- maps the complement of each character in the word.
invWord :: String -> String
invWord = map invChar
```

Also for convenience, I wrote a function that repeats any sequence infinitely many times. This creates an infinite sequence, but since Haskell is a lazy language we can “take” the first n elements out of such a sequence.

```
-- | Repeat a sequence infinitely many times.
--
-- This is a lazy function, so it will not evaluate the
-- sequence infinitely many times.
repeat :: [a] -> [a]
repeat seq = seq ++ repeat seq
```

Finally, we can write our encryption function:

```
-- | Encrypt a word using the Vigenère cipher.
-- NOTE: ‘zipWith’ is a builtin function that takes a function
-- and two sequences and applies the function on
-- corresponding elements in the sequences to generate a new sequence.
encrypt :: String -> String -> String
encrypt text keyword = zipWith shiftChar cleanedText repeatedKeyword
  where
    cleanedText = clean text                     -- drop spaces, punctuation from text
    len = length cleanedText
    repeatedKeyword = take len (repeat keyword) -- get first x letters in sequence
```

And we can define decryption as encryption with the vigenère complement of the key — the respective letters that undo the shifts done during encryption:

```
  -- | Decrypt a word using the Vigenère cipher.
  --
  -- We do the equivalent of encryption with the Vigeère ‘inverse’ of the keyword.
  decrypt :: String -> String -> String
  decrypt text keyword = encrypt text (invWord keyword)
```

Results

```
$ encrypt ‘‘Why is a raven like a writing desk’’ ‘‘rabbithole’’
‘‘NHZJATYOGIELJLMTDFTXZNHEMLR’’


$ decrypt ‘‘NHZJATYOGIELJLMTDFTXZNHEMLR’’ ‘‘rabbithole’’
‘‘WHYISARAVENLIKEAWRITINGDESK’’
```

3. Decrypt the following message, which was encrypted using a Vigenère cipher.

```
mgodt beida psgls akowu hxukc iawlr csoyh prtrt udrqh cengx
uuqtu habxw dgkie ktsnp sekld zlvnh wefss glzrn peaoy lbyig
uaafv eqgjo ewabz saawl rzjpv feyky gylwu btlyd kroec bpfvt
psgki puxfb uxfuq cvymy okagl sactt uwlrx psgiy ytpsf rjfuw
igxhr oyazd rakce dxeyr pdobr buehr uwcue ekfic zehrq ijezr
xsyor tcylf egcy
```

(a) Use the method of displacement coincidences to guess the key length.

(b) Use the Kasiski test to give more evidence for your guess for the key length.

(c) Use frequency analysis with the guessed key length to decrypt the message.

[You are encouraged to use a computer.]

Key Length Estimation

After counting displacement coincidences, I found 7 has the highest number of coincidences.

```
1: 7
2: 6
3: 11
4: 11
5: 9
6: 11
7: 15
8: 4
9: 10
10: 12
11: 11
12: 9
13: 12
14: 17    -- could this be because it is a multiple of 7?
15: 10
16: 6
17: 11
18: 11
19: 7
```

Kasiski Test

I wrote a program that analyzes the recurrences of n-grams in the text.

```
--- 3-grams
*VigenereCipher> run 3
awl: [26,117]       difference: 91
ehr: [227,241]      difference: 14
gki: [61,152]       difference: 91
gls: [12,173]       difference: 161
lsa: [13,174]       difference: 161
psg: [10,150,185]   difference: [140, 175, 35]
sgl: [11,84]        difference: 73
tps: [149,191]      difference: 42
uxf: [156,160]      difference: 4
wlr: [27,118,181]   difference: [91, 154, 63]

--- 4-grams
*VigenereCipher> run 4
awlr: [26,117]        difference: 91
glsa: [12,173]        difference: 161
```

With a length of 3, we see that several n-grams recur in the encrypted message.
Per the **Kasiski Test**, most of the differences in position of repeated n-grams should be multiples of the key-length (7 in this case). We see that $\{14, 35, 42, 63, 91, 140, 154, 161, 175\}$ are all multiples of 7. Only $\{4, 73\}$ are not multiples of 7.

Frequency Analysis

Looking at the highest frequencies over each zeroth, first, second, third, fourth, fifth, and sixth letter modulo 7:

```
1   [ 'i': 15.789473684210526,
      'e': 10.526315789473685
      's': 10.526315789473685
      'l': 7.894736842105263
      'r': 7.894736842105263
      ...
    ],
2   [ 'r': 15.789473684210526
      'a': 10.526315789473685
      'y': 10.526315789473685
      'b': 7.894736842105263
      'e': 7.894736842105263
      ...
    ],
3   [ 'u': 18.42105263157895
      'k': 15.789473684210526
      'o': 13.157894736842104
      't': 7.894736842105263
      'z': 7.894736842105263
      ...
    ],
4   [ 'p': 13.157894736842104
      'c': 10.526315789473685
      'e': 10.526315789473685
      't': 10.526315789473685
      'w': 7.894736842105263
      ...
    ],
5   [ 's': 13.157894736842104
      'a': 10.526315789473685
      'e': 10.526315789473685
      't': 10.526315789473685
      'c': 7.894736842105263
      ...
    ],
6   [ 'g': 16.216216216216218
      'b': 10.81081081081081
      'u': 10.81081081081081
      'y': 10.81081081081081
      'f': 8.108108108108109
      ...
    ],
7   [ 'l': 13.513513513513514
      'w': 10.81081081081081
      'd': 8.108108108108109
      'h': 8.108108108108109
      'p': 8.108108108108109
      'f': 5.405405405405405
      ...
    ]
```

Suppose the keyword $k = k_1k_2k_3k_4k_5k_6k_7$ where k_1 is the first letter of the message, etc.
Since we expect the most common letters to have similar recurrence across the text, we can pick out one recurring frequency (15.789473684210526) and check the values closest to that frequency.
We can expect that:

$$\exists p_i \in \mathcal{P} \ni p_i \begin{cases} \underset{k_1}{\Rightarrow} \text{`i'} & \in \mathcal{C} \\ \underset{k_2}{\Rightarrow} \text{`r'} & \in \mathcal{C} \\ \underset{k_3}{\Rightarrow} \text{`k'} & \in \mathcal{C} \\ \underset{k_4}{\Rightarrow} \text{`p'} & \in \mathcal{C} \\ \underset{k_5}{\Rightarrow} \text{`s'} & \in \mathcal{C} \\ \underset{k_6}{\Rightarrow} \text{`g'} & \in \mathcal{C} \\ \underset{k_7}{\Rightarrow} \text{`l'} & \in \mathcal{C} \end{cases}$$

From the above, we can guess that:

$$\begin{aligned}
\textbf{let } k_1 &\equiv \text{‘A’}\\
k_2 - k_1 = \text{‘r’} - \text{‘i’} = 9 \Rightarrow k_2 &\equiv \text{‘J’}\\
k_3 - k_1 = \text{‘k’} - \text{‘i’} = 2 \Rightarrow k_3 &\equiv \text{‘C’}\\
k_4 - k_1 = \text{‘p’} - \text{‘i’} = 7 \Rightarrow k_4 &\equiv \text{‘H’}\\
k_5 - k_1 = \text{‘s’} - \text{‘i’} = 22 \Rightarrow k_5 &\equiv \text{‘W’}\\
k_6 - k_1 = \text{‘g’} - \text{‘i’} = 24 \Rightarrow k_6 &\equiv \text{‘Y’}\\
k_7 - k_1 = \text{‘l’} - \text{‘i’} = 3 \Rightarrow k_7 &\equiv \text{‘D’}
\end{aligned}$$

We can now run a brute-force shift cipher attack on the relations of the keyword, and try to look for a recurring pattern.

```
*ShiftCipher> bruteforce ‘‘AJCHWYD’’          13: nwpujlq
0: ajchwyd                                    14: mvotikp
1: zibgvxc                                    15: lunshjo
2: yhafuwb                                    16: ktmrgin
3: xgzetva                                    17: jslqfhm
4: wfydsuz                                    18: irkpegl
5: vexcrty                                    19: hqjodfk
6: udwbqsx                                    20: gpincej
7: tcvaprw                                    21: fohmbdi
8: sbuzoqv                                    22: englach
9: ratynpu                                    23: dmfkzbg
10: qzsxmot                                   24: clejyaf
11: pyrwlns                                   25: bkdixze
12: oxqvkmr
```

Much of the results doesn’t make sense (as we expected), but one shift almost spells “England”.
Let’s focus on the possibility of that being our keyword — in which case the last two characters we picked are likely wrong.

Using ‘‘ENGLAND’’ as the keyword, we get the following results:

```
ITISTOBEQUESTIONEDWHETHERINTHEWHOLELENGTHANDBREADTHOF
THEWORLDTHEREISAMOREADMIRABLESPOTFORAMANINLOVETOPASS
ADAYORTWOTHANTHETYPICALENGLISHVILLAGEITCOMBINESTHE
COMFORTSOFCIVILIZATIONWITHTHERESTFULNESSOFSOLITUDE
INAMANNEREQUALLEDBYNOOTHERSPOTEXCEPTTHENEWYORKPUBLICLIBRARY
```

When we space out and format the text properly, it reads:

It is to be questioned whether in the whole length and breadth of the world there is a more admirable spot for a man in love to pass a day or two than the typical English village. It combines the comforts of civilization with the restfulness of solitude in a manner equalled by no other spot except the New York public library.

4. Consider the quadratic map

$$E : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$$
$$x \mapsto x^2 + ax + b$$

with $a, b \in \mathbb{Z}/n\mathbb{Z}$. Show that if $n \neq 2$, then E is *never* an encryption function (i.e., E cannot be inverted). What can you say about other maps $x \mapsto f(x)$ where $f(x) \in \mathbb{Z}[x]$, in particular, are any polynomial maps of higher degree invertible?

An encryption function has to be invertible (for decryption), and invertible functions must be injections and a surjections (i.e. bijections). These conditions are not satisfied by the quadratic map $E : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z} \ni x \mapsto x^2 + ax + b$ unless we restrict its domain and range to $\mathbb{Z}/2\mathbb{Z}$. First, for a multiplicative map over $\mathbb{Z}/n\mathbb{Z}$ to be invertible, $\mathbb{Z}/n\mathbb{Z}$ must be a field — otherwise some elements in $\mathbb{Z}/n\mathbb{Z}$ will not have multiplicative inverses.

Consider some finite field $\mathcal{F}_n$ formed by taking the integers modulo n, where n is a prime number. As an example, we'll take $\mathcal{F}_5$

Every element in $\mathcal{F}$ except 0 has a multiplicative inverse and an additive inverse. For example: Let's define $\mathcal{F}_5 = \{0, 1, 2, 3, 4\}$ Then, we can define multiplicative inverses and additive inverses in $\mathcal{F}_5$ as:

f_i	f_i^{-1}	$-f_i$
1	1	4
2	3	3
3	2	2
4	4	1

As in all number systems (that I know of. . . I'd like to know of exceptions), $f_i^2 = (f_i^{-1})^2$.

Similarly, $f_i + (-f_i) = 0$

Examples in F_5:

$$3^2 \equiv 9 \pmod 5 \equiv 4 \equiv 2^2$$
$$3 + 2 \equiv 5 \pmod 5 \equiv 0$$

Hence. . . the quadratic map $E : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z} \ni x \mapsto x^2 + ax + b$ is guaranteed to not be surjective under these conditions:

(a) If $a = 0$, then $E(x^{-1}) \equiv x^2 + b \equiv E(x)$
(b) Generally, $E(-a) \equiv a^2 - a^2 + b \equiv b \equiv E(0)$

However, if we limit the domain and range to $\mathcal{F}_2$, then the only non-zero element is 1, and:

(a) If $a = 0$, then $E(x^{-1}) \equiv x^2 + b \equiv E(x)$, but $1 \equiv 1^{-1} \pmod 2$
(b) $-1 \equiv 1 \pmod 2$, and $E(-1) = E(1) = 1 + a + b \not\equiv E(0)$

In general, it is possible for higher-order polynomials to be invertible, but that property does not hold for *all* higher-order polynomials. Careful thought should be put in choosing a polynomial as an encryption function (or, just use modern techniques?).

For instance, $x \mapsto x^3$ (and any such function as $x \mapsto x^{\text{odd power}}$, it seems) is invertible in any finite field, but $x \mapsto x^4$ (and any other $x \mapsto x^{\text{even power}}$ function) is not.

5. Let $D_n = \{x \in \mathbb{R}^n : \sum_{i=1}^n x_i^2 = 1\}$ be the unit sphere in $\mathbb{R}^n$. Fix $x \in D_n$ and consider the function $\psi_x : D_n \to \mathbb{R}$ defined by

$$\psi_x(y) = x \cdot y = \sum_{i=1}^n x_i y_i.$$

Show that the function ψ_x achieves a unique maximum at $x = y$. How does this relate to frequency analysis?

Let:

$$\vec{x} = \langle x_1, x_2, \ldots, x_n \rangle \ni \sum_{i=1}^n x_i^2 = 1 \Rightarrow |\vec{x}| = 1$$

$$\vec{y} = \langle y_1, y_2, \ldots, y_n \rangle \ni \sum_{i=1}^n y_i^2 = 1 \Rightarrow |\vec{y}| = 1$$

$$\vec{x} \cdot \vec{y} = |\vec{x}|\,|\vec{y}| \cos\theta \leq |\vec{x}|\,|\vec{y}| \quad \text{(since } \cos\theta \leq 1\text{)}$$

To achieve a maximum:

$$\underbrace{\cos\theta \leq 1}_{\text{maximize this}} \Rightarrow \cos\theta = 1$$

$$\angle(\vec{x}, \vec{y}) = \arccos 1 = 0$$

Thus, we know that $\vec{x}$ and $\vec{y}$ are the same vector, since $\angle(\vec{x}, \vec{y}) = 0$, and $|\vec{x}| = |\vec{y}| = 1$ (the vectors have the same direction and the same magnitude).

Suppose we have $\vec{f} = \langle f_1, f_2, \ldots, f_n \rangle$ corresponding to the frequencies of the n letters in an alphabet, such that $\sum_{i=1}^n f_i = 1$. Then, this maximum implies that $\vec{f} \cdot \vec{f}$ gives a greater value than $\vec{f} \cdot \vec{f}_{\to k}$ where $\vec{f}_{\to k}$ is the vector obtained by shifting and cycling the elements in $\vec{f}$ to the right by k positions. On the vigenère cipher, for instance, this means that the frequencies (or coincidences, an approximation for frequencies) will be maximized when the shift matches or is a multiple of the key-length.

Challenge problem: (Try it for fun, you are not required to submit written-up solutions, unless you are a graduate student enrolled in the class.)

6. Let $n, k \in \mathbb{Z}_{>0}$ and recall the general linear group $\mathrm{GL}_k(\mathbb{Z}/n\mathbb{Z})$.

(a) Write down all the elements of $\mathrm{GL}_2(\mathbb{Z}/2\mathbb{Z})$. What more commonly known group is this isomorphic to?

(b) If $n = p$ is a prime number, prove that $\mathrm{GL}_k(\mathbb{Z}/p\mathbb{Z})$ has $(p^k - 1)(p^k - p) \cdots (p^k - p^{k-1})$ elements. *[Use linear algebra over the field $\mathbb{Z}/p\mathbb{Z}$ and think of building your matrix one column at a time.]*

(c) Prove that if n, m are relatively prime positive integers, then

$$\#\mathrm{GL}_k(\mathbb{Z}/nm\mathbb{Z}) = \#\mathrm{GL}_k(\mathbb{Z}/n\mathbb{Z}) \cdot \#\mathrm{GL}_k(\mathbb{Z}/m\mathbb{Z}).$$

The following subparts will provide a guide to an algebraic proof of this fact (not all of these require a proof, they are a kind of series of hints to guide your work).

(a) For n, m relatively prime, the map $\phi : \mathbb{Z}/nm\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$, defined by $a \mapsto (a \bmod n, a \bmod m)$, is an isomorphism of groups. We can write $\phi(a) = (\phi_n(a), \phi_m(a))$ where $\phi_n : \mathbb{Z}/nm\mathbb{Z} \to \mathbb{Z}/n\mathbb{Z}$ is the reduction modulo n homomorphism and similarly for ϕ_m. In fact, ϕ is an isomorphism of rings with 1, i.e., respects multiplication and the multiplicative identity.

(b) Promote ϕ to an isomorphism $\Phi : M_k(\mathbb{Z}/nm\mathbb{Z}) \to M_k(\mathbb{Z}/n\mathbb{Z}) \times M_k(\mathbb{Z}/m\mathbb{Z})$ of rings with 1 by sending a matrix $A = (a_{ij})_{1 \le i,j \le k}$ to the pair $(\Phi_n(A), \Phi_m(A))$, where $\Phi_n(A) = (\phi_n(a_{ij}))_{1 \le i,j \le k}$ is the result of reducing all entries of A modulo n, and similarly for $\Phi_m(A)$. First you have to prove that Φ is a ring homomorphism, then that it is injective and surjective, which relies crucially on the injectivity and surjectivity of ϕ.

(c) Prove that $\phi(\det(A)) = (\det(\Phi_n(A)), \det(\Phi_m(A)))$ for all $A \in M_k(\mathbb{Z}/nm\mathbb{Z})$. Colloquially, this says that ϕ and Φ "respect" the determinant.

(d) Prove that $A \in M_k(\mathbb{Z}/nm\mathbb{Z})$ is invertible if and only if $\Phi(A)$ is an invertible element of the ring $M_k(\mathbb{Z}/n\mathbb{Z}) \times M_k(\mathbb{Z}/m\mathbb{Z})$ if and only if both $\Phi_n(A) \in M_k(\mathbb{Z}/n\mathbb{Z})$ and $\Phi_m(A) \in M_k(\mathbb{Z}/m\mathbb{Z})$ are invertible. Conclude that Φ induces a group isomorphism $\mathrm{GL}_k(\mathbb{Z}/nm\mathbb{Z}) \cong \mathrm{GL}_k(\mathbb{Z}/n\mathbb{Z}) \times \mathrm{GL}_k(\mathbb{Z}/m\mathbb{Z})$ and as a consequence, we get the desired formula.

(d) Recall the affine cipher with $\mathcal{P} = \mathcal{C} = (\mathbb{Z}/n\mathbb{Z})^k$ and with key $A \in \mathrm{GL}_k(\mathbb{Z}/n\mathbb{Z})$. If Eve discovers the encryption of k plaintext elements, prove that the probability that she can solve for the key is $\#\mathrm{GL}_k(\mathbb{Z}/n\mathbb{Z})/n^{k^2}$. Compute this probability for $n = 26$ and $k = 2, 3, 4$. *[This was done a bit too quickly in lecture, so check it yourself.]*

(e) After experimenting, what can you say about this probability as $k \to \infty$ or as $n \to \infty$?