8 Cryptography

8.1 Character Ciphers

---

1. **Introduction:** The goal of this entire chapter (and the rest of the course) is to talk about encryption and cryptography.
2. **Terminology:** We have the following:
   (a) *Cryptology*: The study of encryption/decryption.
   (b) *Cryptography*: The study of methods of encryption/decryption.
   (c) *Cipher*: A particular method of encryption.
   (d) *Cryptanalysis*: Breaking of systems of encryption.
   (e) *Plaintext*: The human-readable text we wish to encryp.
   (f) *Encryption*: The process of applying a cipher to plaintext.
   (g) *Ciphertext*: The human-non-readable result.
   (h) *Decryption*: The process of getting the plaintext back.
   (i) *Some Names*:
      i. Alice: encrypts and sends
      ii. Bob: receives and decrypts
      iii. Eve: eavesdropper
3. **Basic Methods:**
   (a) **Character Assignment:** To begin, we will assign a number to each letter of the alphabet:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

   Note: For now we will exclude lower-case, punctuation and spaces, but we could include those and use a different modulus.
   Note: This can be confusing since `A` is the first leter of the alphabet and so we would naturally want to assign it to 1. We use this for purposes of making our modular arithmetic easier.

   (b) **Shift Cipher:** For each plaintext letter P we assign ciphertext

$$C \equiv P + b \bmod 26$$

Ex. Encrypt `LEIBNIZ` with $b = 3$.

$$\begin{aligned}
L: &\quad P = 11, 11 + 3 \equiv 14 = C : O \\
E: &\quad P = 4, 4 + 3 \equiv 7 = C : H \\
I: &\quad P = 8, 8 + 3 \equiv 11 = C : L \\
B: &\quad P = 1, 1 + 3 \equiv 4 = C : E \\
N: &\quad P = 13, 13 + 3 \equiv 16 = C : Q \\
I: &\quad P = 8, 8 + 3 \equiv 11 = C : L \\
Z: &\quad P = 25, 25 + 3 \equiv 2 = C : C
\end{aligned}$$

Which then results in `OHLEQLC`. To decrypt we simply reverse: $C \equiv P + b \bmod 26$, $P \equiv C - b \bmod 26$.

(c) **Affine Cipher:** Choose a and b and encrypt via $C = aP + b \bmod 26$. How will decryption work? $C \equiv aP + b \bmod 26$, $aP \equiv C - b \bmod 26$ there needs to be a unique P. To have this we need $\gcd(a, 26) = 1$ so that a has a multiplicative inverse. Then $P \equiv a^{-1}(C - b) \bmod 26$. How many choices? $\phi(26) = 12$ for a and 26 choices for b.

Ex. If we choose $a = 5$ and $b = 7$ then encryption is $C \equiv 5P + 7 \bmod 26$ and decryption is $5P \equiv C - 7 \bmod 26 \implies P \equiv 21(C - 7) \bmod 26$ (calculated from 21 being the multiplicative inverse of 5).

4. **Breaking Shift Ciphers:** To break a shift cipher, we only need b. For example, if we manage to find a specific C_0 for a specifice P_0, then we know that $C_0 \equiv P_0 + b \bmod 26$ so $b \equiv C_0 - P_0 \bmod 26$. How might we do this? With frequency analysis.

   Frequency Analysis: In english, the most frequent letter is `E`, note this is $P_0 = 4$. Find the most frequent ciphertext letter. If that is C_0 we guess at that.

5. **Breaking Aphine Ciphers:** One C_0 and P_0 pair is not sufficient! Since knowing $C_0 \equiv aP_0 + b \bmod 26$ is not enough to find a and b. However, having another pair is good enough because:

$$\begin{aligned}
C_0 &\equiv aP_0 + b \bmod 26 \\
C_1 &\equiv aP_1 + b \bmod 26 \\
\hline
C_0 - C_1 &\equiv a(P_0 - P_1) \bmod 26
\end{aligned}$$

   This will have solutions if and only if $\gcd(P_0 - P_1, 26) \mid C_0 - C_1$, and if so there will be $\gcd(P_0 - P_1, 26)$ solutions.

   Note: Keep in mind this is valid cipher text. There is an a (which Alice chose). So there will be solutions. There may be more than 1. If multiple possible a, for each, find b, simply try all of those a, b combinations until we get proper plaintext.

8.2 Exponentiation Ciphers

---

1. **Introduction:** Can we find a process which is harder to invert?
First we will modify the table of letters slightly:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

Now, we can put letters together unambigiously. For example `JU` can be assigned to 0920 or just 920. Without the leading 0 it is unclear what something like 111 means. It could be $111 \implies 0111$ or $111 \implies 1101$.

Fermat's Little Theorem: Recall, if p is prime and $a \in \mathbb{Z}$ with $p \nmid a$ then $a^{p-1} \equiv 1 \bmod p$.

2. **Exponentiation Cipher**

   (a) **Encryption:** Let p be an odd prime (typically very large) and let e be a positive integer with $\gcd(e, p-1) = 1$ (use Euclidean Algorithm for this). We then take the plaintext and group the letters into blocks so no block is larger than p.
   For example,

   - If $p = 29$ then blocksize is 1 since `z` $\leftrightarrow 25 < p$.
   - If $p = 3001$ then blocksize is 2 since `zz` $\leftrightarrow 2525 < p$.
   - If $p = 377173$ then blocksize is 3 since `zzz` $\leftrightarrow 252525 < p$.

   We then pad the plaintext with junk letters at the end if needed so that the plaintext length is a multiple of the blocksize. Traditionally `X` is used but any letter can be used. To encrypt, Alice needs to divide full plaintext into blocks. For each block P we do

   $$C \equiv P^e \bmod p$$

   Ex. Alice wants to encryp `LOVENOTE` with $(e, p) = (479, 3001)$ and $\gcd(479, 3000) = 1$.

   $$\begin{array}{c|llll} & \texttt{LO} & \texttt{VE} & \texttt{NO} & \texttt{TE} \\ & 1114 & 2104 & 1314 & 1904 \\ & 1114^{479} & 2104^{479} & 1314^{479} & 1904^{479} \\ \hline \equiv & 0169 & 0317 & 0017 & 1697 \end{array}$$

   So we get `0169 0317 0017 1697` as the ciphertext that Alice would send to Bob.

(b) **Decryption:** This process is invertible since the fact that $\gcd(e, p-1)$ guarantees that there exists some d with $de \equiv 1 \bmod p$. Then for a ciphertext block raised to d:

$$C^d \equiv (P^e)^d \equiv P^{ed} \equiv P^{1+k(p-1)} \equiv P(P^{p-1})^k \equiv P(1)^k \equiv P \bmod p$$

Here the fact that $P^{p-1} \equiv 1 \bmod p$ is guaranteed by FLiT. Note that $p \nmid P$ since $P < p$.

Thus, to decrypt ciphertext, Bob simply takes C and raises it to d, $C^d \bmod p$.