Brandon Hang - Assignment # 3

1. (3 marks) Let K be a fixed element of Z_{26}, and define the permutation π over Z_{26} as

$$\pi = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 \\ 23 & 13 & 24 & 0 & 7 & 15 & 14 & 6 & 25 & 16 & 22 & 1 & 19 & 18 & 5 & 11 & 17 & 2 & 21 & 12 & 20 & 4 & 10 & 9 & 3 & 8 \end{pmatrix}.$$

Consider the stream cipher with the key stream element $z_i \in Z_{26}$, where $z_i = (K + i - 1) \bmod 26$. Encryption and decryption using π are done as follows:

$$e_z(x) = (\pi(x) + z) \bmod 26,$$

$$d_z(y) = (\pi^{-1}(y - z) \bmod 26).$$

Write a C/C++ program streamCipher to decrypt the following ciphertext which was encrypted by the above stream cipher:

WRTCNRLDSAFARWKXFTXCZRNHNY PDTZUUKMPLUSOXNEUDOKLXRMCBKGRCCURR

Your program should use brute force attack and stop when the user is satisfied with the result. (submit the plaintext, your code, and instructions on how to test it).

After the 11th iteration, the plain text that makes the most sense is:
"THEFIRSTDEPOSITCONSISTEDOFONETHOUSANDANDFOURTEENPOUNDSOFGOLD".

Explanation:

1. The key stream 'zi' is calculated as (k + I – 1) mod 26 where k is 26 and I is the index of the key stream.
2. We have to find the inverse of permutation π which will decrypt the ciphertext by trying all possible k values (0 - 25).
3. For each key, it will calculate the key stream 'z' and prints each decrypted character until the length of the cipher text has gone through.
4. After the decrypted message is displayed, it will prompt the user whether they want to continue or not (as in use the next key). This will help decide whether the decrypted message makes sense or not.

The program demonstrates a brute force attack by trying all of the possible keys. The C++ code is submitted along side with this document.

Note: the solution for this exercise will not be posted to blackboard.

2. (3 marks) Encrypt the message meet me at the usual place at ten rather than eight oclock using the Hill cipher with the key

$$K = \begin{bmatrix} 9 & 4 \\ 5 & 7 \end{bmatrix}$$

Show your calculations (for at least one pair of the plaintext) and the corresponding decryption of the ciphertext to recover the original plaintext.
Note: you may append arbitrary letter to make the length of the plaintext a multiple of m.

To encrypt the message, we first grab a pair of characters in the message. The characters are converted into a numerical value where a=0, b=1, c=2 … z=25. The first 2 character are "m" and "e" so m = 12 and e = 4. The value are in a vector and is multiplied by the key

and mod 26 to maintain the range of the alphabet:

$$\begin{bmatrix} 9 & 4 \\ 5 & 7 \end{bmatrix} \begin{bmatrix} 12 \\ 4 \end{bmatrix} = \begin{bmatrix} 124 \\ 88 \end{bmatrix} \text{Mod} 26 = \begin{bmatrix} 20 \\ 10 \end{bmatrix}$$

From numerical value 20 and 10, the characters are 'u' and 'k' which is replaced by the characters 'm' and 'e'. So the first iteration of the encrypted plain text will be "uket me at the usual place at ten rather than eight oclock" where the first 2 character are encrypted. Repeat by grabbing the next pair until rest of the plain text is encrypted. The encrypted message is: "ukix uk yd rom eiwsz xwiok un ukh xhroaj roan qyebt lkjegiR" with the given key.

3. (4 marks) Develop an algorithm to decrypt the ciphertext below which was encrypted using a Hill cipher with $m = 2$:

LMQETXY EAGTXCTUIEWNCTXLZEWUAISPZY V APEWLMGQWY A
XFTCJMSQCADAGTXLMDXNXSNPJQSY V APRIQSMHNOCV AXFV

Note: use digram frequencies. The 10 digrams with highest frequencies in English are (in the order of decreasing frequency) TH, HE, IN, ER, AN, RE, ED, ON, ES, ST.

1.) Divide the cipher text into 2 letter digrams in a list. ["LM", "QE", "TX", … "FV"]
2.) Count the frequency of each diagram in the cipher text.
3.) Rank the digrams based on the highest frequency and compare it with the 10 digrams with the highest frequencies in English.
4.) Assign the most frequent diagram in the cipher text to the most common English digrams (TH) and the second most frequent diagram to the second common English diagram (HE).
5.) By assigning those letters, we can translate them into a Unicode value which will give a numerical value for each letter.
6.) We then find the key values by solving the systems of equation where X1 = first char of first English bigram, Y1 = second char of first English bigram, X2 = First char of second English bigram, Y2 = Second char of second English bigram, C1 = first char of first high freq bigram, C2 = second char of first high freq bigram, C3 = first char of second high freq bigram, C4 = second char of second high freq bigram.
    X1 * k00 + Y1 * k01 = C1
    X1 * k10 + Y1 * k11 = C2
    X2 * k00 + Y2 * k01 = C3
    X2 * k10 + Y2 * k11 = C4
7.) Find the inverse of the key matrix that was generated above multiply it with the vectors of the cipher text pairs.
8.) Repeat until all the cipher text bigram pairs are translated into the plain text.
9.) Repeat 4-8 until the frequency of the diagrams in the ciphertext is similar to that of the English language.

The Algorithm was implemented in the Hill Cipher attack lab where the program was implemented in python decrypting the message to be "thekingwasinhiscountinghousecountingouthismoneythequeenwasintheparloureatingbreadandhoneyz" and the key being [[4 13], [11, 9]].

In our next class meeting, we will have a ip class lab activity. You will write a C/C++ program hillCipher to implement your algorithm. Design the program to try the more likely possibilities rst. Your program should work interactively with the user (show potential

plaintext after each try) and stop when the user is satis ed with the result. Note: the solution for this exercise will not be posted to blackboard.