SIL765: Networks and System Security

Assignment - 01

under the guidance of

Prof. Vireshwar Kumar

Submitted by:

Harish Yadav (2021CSY7544)

Indian Institute of Technology, Delhi

Semester II, 2021-22

Description and Approach

- Provide cipher text to be cryptanalyzed and decrypted in **ciphertext.txt** file in the same folder in which the decipher_text.cpp is present in.
- Use **`$make`** command to execute the program. It compiles decipher_text.cpp, and create an executable file named decipher_text.cpp and executes the executable file.

Includes/CharIntegerPairVectorOperations.h

- Method **cIPairVectorValCmp()** returns true if first argumented pair's second element is greater than second argumented pair's second element. It is used for sorting the vector of pair of char and integer based on the second element in non-increasing order.
- Method **findCIPairVector()** returns the iterator to pair of char and integer in the argumented vector having first element equal to argument key.

Includes/StringIntegerPairVectorOperations.h

- Method **sIPairVectorValCmp()** returns true if first argumented pair's second element is greater than second argumented pair's second element. It is used for sorting the vector of pair of string and integer based on the second element in non-increasing order.
- Method **findSIPairVector()** returns the iterator to pair of string and integer in the argumented vector having first element equal to argument key.

Includes/StringVectorOperations.h

- Method **findStringVector()** returns true if argumented string belongs to the argumented vector of string

Includes/CommonWordsDictionary.h

- Contains around 500 common words used in spoken English and each of length 5 to 10 in vectors of strings.

Includes/CipherCharacterOperations.h

- Method **isCipherCharacter()** returns true if the argument character belongs to the enlisted cipher characters
- Method **getCipherCharacterIndex()** returns the index of argumented cipher text as per the predefined indices for cipher characters
- Method **getCipherCharacterFromIndex()** returns the character of argumented index as per the predefined indices for cipher characters
- The enlisted cipher characters along with their defined indices are as follows:

Cipher Character	Index
n – z	0 - 12
0 – 9	13- 22
@	23
#	24
$	25

decipher_text.cpp

- This file contains the main() function which is responsible for cryptanalysis of the cipher text provided using Heuristics of English language and later using Dictionary attack.
- The algorithm/flow of program is as follows:
  - First input the ciphertext from the file **ciphertext.txt** which is present in the same folder as decipher_text.cpp
  - Then evaluate and store frequencies of different properties like frequency of:
    - each cipher characters
    - first letter of words
    - last letter of words
    - digraphs
    - trigraphs
    - doubles
    - one letter words

        - two letter words
        - three letter words
        - four letter words
    - Then sort these frequencies in non-increasing order
    - Apply Heuristics of English in order to evaluate some of the substitutions. The heuristics applied in this program are:
        - In English, there are only 2 one-letter English words, and they are ‘a’ and ‘I’. Out of which frequency of ‘a’ is quite very high than ‘I’ and hence, we use this knowledge to substitute the most occurring one-letter cipher word to be ‘a’ and in case if another one-letter word exists, then we substitute it to ‘i’.
        - Using ‘a’ as an anchor for 'and' which is one of the most frequent trigraph in English. First check if 'a' is substituted to any cipher text character or not, then find most frequent three-letter word in given cipher-text with first cipher character as one substituted for 'a' and second and third characters not being equal, as it would not lead to 'and' rather to 'all' according to heuristics of English language. For confirmation, just checking whether it is also one of the most frequent trigraph as 'are' is one of the most frequent three-letter word but not most frequent trigraph. Then substituting most frequent trigraph's (starting with character substituted for 'a') second cipher character with 'n' and third cipher character with 'd'.
        - Using 'a' as an anchor and 'and' if recovered for 'are' which is one of the most frequent three-letter word in English. Check if 'a', 'n' and 'd' are substituted to any cipher text character or not. Then find most frequent three-letter word in given cipher-text with first cipher character as one substituted for 'a' and second and third characters not being equal, nor being substituted to 'n' and 'd' respectively. Ultimately, substitute most frequent three-letter's (starting with character substituted for 'a' and not ending with 'nd' or double) second cipher character with 'r' and third cipher character with 'e'
        - Using 'all' being one of the most frequent three-letter word to find 'l' if 'a' is recovered. First check if 'a' is recovered, if yes then substitute most frequent three-letter word's (with first character substituted for 'a' and

same second and third character) second and third cipher character with 'l'.

    - Using 'th', 'the' and 'the' being one of the most frequent bigraph, trigraph and three letter word to find 't', 'h', 'e'. For this check if the chosen three-letter word is not 'and' as it also resembles the property along with 'an'. Then substitute most frequent bigraph, trigraph, three-letter word's first cipher character with 't', second cipher character with 'h' and third cipher character with 'e'.
    - Utilizing 'to' being one of the most frequent two letter word to find 'o' if 't' is recovered. Then substituting most frequent two-letter word's (starting with character substituted for 't') second cipher character with 'o'.
    - 'any' being one of the most frequent three letter word can be utilized find 'y' if 'a', 'n' and 'd' are recovered. For this check if 'a', 'n' and 'd' are recovered, then substitute most frequent three-letter word's (starting with characters substituted for 'an' and third character not being substituted for 'd') third cipher character with 'y'.
    - Using 'with' being one of the most frequent four-letter word 'w' and 'i' substitutions can be recovered if 't' and 'h' are recovered by substituting most frequent four-letter word's (ending with character substituted for 'th') first cipher character with 'w' and second cipher character with 'i'.
    - 'for' being one of the most frequent three-letter word can be utilized to find 'f' and 'r' if 'o' is recovered by substituting most frequent three-letter word's (with middle character substituted for 'o') first cipher character with 'f' and third cipher character with 'r'.
- Then apply Dictionary Attack to evaluate rest of the substitutions. The dictionary attack in this program in following manner:
    - First segment words as per their lengths (to keep the confusion to be minimal, in this program words with length 5 to 10 are used for dictionary attack, as if word with length less than 5 is used then it would led to miscomputation of substitutions. eg. is ‘u’, ‘s’, ‘t’ are evaluated then a 4 letter word would look like ‘*ust’ now * could be substituted with ‘m’, ‘j’, ‘r’, etc. and hence to keep mispredictions to be minimum, letter of length 5 to 10 are used for dictionary attack.

        - Then do the following attack either till updates are being made or till we did find all the cipher characters in the given cipher text:
            - Perform the following attack on cipher words of length 5 to 10:
                - Find the number of characters which are already substituted. Check if number of substituted words are greater than half the character, just to be sure that a wrong word for the given cipher text word is considered.
                - Check the compatibility of dictionary word along with the cipher text word.
                - If word proves to be compatible, then substitute words accordingly, by substituting the non-substituted plain-text characters with the corresponding cipher-text characters according to dictionary word and cipher-text word.
    - Handle non-substituted cipher characters (Since, this program is highly dependent over the Heuristics for English language and also the dictionary of common words used in English, there are chances where all the cipher text characters present in the cipher-text are not substituted. Hence, we just substitute the cipher-text characters for the remaining characters which are not substituted, either due to limitations of the program or due to absence in the cipher-text character). This step is taken just to keep uniformity of the deciphered text.
    - Evaluate Plaintext based on the Deciphered key:
        - For each character of ciphertext, check whether the character belongs to the ciphertext character set or not.
            - If yes, then substitute it with the corresponding plaintext character according to the deciphered key.
            - Else, let the character be.
    - Print the ciphertext, deciphered plaintext and deciphered key.
    - Output the deciphered plaintext and deciphered key in **deciphered_ plaintext.txt** and **deciphered_key.txt** respectively.

Heuristics used for Deciphering

Letter	Heuristic 1	Heuristic 2
a	most frequent one letter word	
b		
c		
d	using 'and' and 'a' to find 'd'	
e	using 'th', 'the' and 'the' to be the most frequent bigraph, trigraph and three letter word to find 'e'	using 'a' and 'are' to find 'e'
f	using 'o' and 'for' to find 'f'	
g		
h	using 'th', 'the' and 'the' to be the most frequent bigraph, trigraph and three letter word to find 'h'	
i	second most frequent one letter word	using 't', 'h' and 'with' to find 'i'
j		
k		
l	using 'a' and 'all' to find 'l'	
m		
n	using 'and' and 'a' to find 'n'	
o	using 'to' and 't' to find 'o'	
p		
q		
r	using 'a' and 'are' to find 'r'	using 'o' and 'for' to find 'r'
s		
t	using 'th', 'the' and 'the' to be the most frequent bigraph, trigraph and three letter word to find 't'	
u		
v		
w	using 't', 'h' and 'with' to find 'w'	
x		
y	using 'a', 'n' and 'any' to find 'y'	
z		

Result

- **Ciphertext 1**

  Ciphertext (Input):

  1981y, $pp1n1yuux oq@ 2@3s5u1n $p 1981y, 1v y n$s9o2x 19 v$soq yv1y. 1o 1v oq@ v@6@9oq uy27@vo n$s9o2x 5x y2@y, oq@ v@n$98 0$vo 3$3su$sv n$s9o2x, y98 oq@ 0$vo 3$3su$sv 8@0$n2ynx 19 oq@ #$2u8. 5$s98@8 5x oq@ 1981y9 $n@y9 $9 oq@ v$soq, oq@ y2y51y9 v@y $9 oq@ v$soq#@vo, y98 oq@ 5yx $p 5@97yu $9 oq@ v$soq@yvo, 1o vqy2@v uy98 5$28@2v #1oq 3yw1voy9 o$ oq@ #@vo; nq19y, 9@3yu, y98 5qsoy9 o$ oq@ 9$2oq; y98 5y97uy8@vq y98 0xy90y2 o$ oq@ @yvo. 19 oq@ 1981y9 $n@y9, 1981y 1v 19 oq@ 61n191ox $p v21 uy9wy y98 oq@ 0yu816@v; 1ov y98y0y9 y98 91n$5y2 1vuy98v vqy2@ y 0y21o10@ 5$28@2 #1oq oqy1uy98, 0xy90y2 y98 198$9@v1y. 7$$8, 9$# os29 p$2 oq@ v@n$98 3y2o $p oq@ 4s@vo1$9, 7$$8 usnw!

  Deciphered Plaintext:

  india, officially the republic of india, is a country in south asia. it is the seventh largest country by area, the second most populous country, and the most populous democracy in the world. bounded by the indian ocean on the south, the arabian sea on the southwest, and the bay of bengal on the southeast, it shares land borders with pakistan to the west; china, nepal, and bhutan to the north; and bangladesh and myanmar to the east. in the indian ocean, india is in the vicinity of sri lanka and the maldives; its andaman and nicobar islands share a maritime border with thailand, myanmar and indonesia. good, now turn for the second part of the question, good luck!

  Deciphered Key:

  y5n8@p7q1rwu09$342vos6#txz

- **Ciphertext 2**

Ciphertext (Input):

64s48u46 8y6 q480ryp nrv 6ryy43 2yu$2tn46, n4 54yu u$ o46. un8u yrpnu n4 6r6 y$u vq441 54qq, n80ryp s4043rvn 6348wv, n80ryp y$ 34vu. n4 58v 2yv234 5n4un43 n4 58v 8vq441 $3 6348wryp. t$yvtr$2v, 2yt$yvtr$2v, 8qq 58v 8 oq23. n4 34w4wo4346 t3#ryp, 5rvnryp, n$1ryp, o4ppryp, 404y q82pnryp. n4 sq$8u46 un3$2pn un4 2yr043v4, v44ryp vu83v, 1q8y4uv, v44ryp 483un, 8qq o2u nrwv4qs. 5n4y n4 q$$z46 6$5y, u3#ryp u$ v44 nrv o$6#, un434 58v y$unryp. ru 58v x2vu un8u n4 58v un434, o2u n4 t$2q6 y$u s44q 8y#unryp s$3 x2vu nrv 134v4yt4.

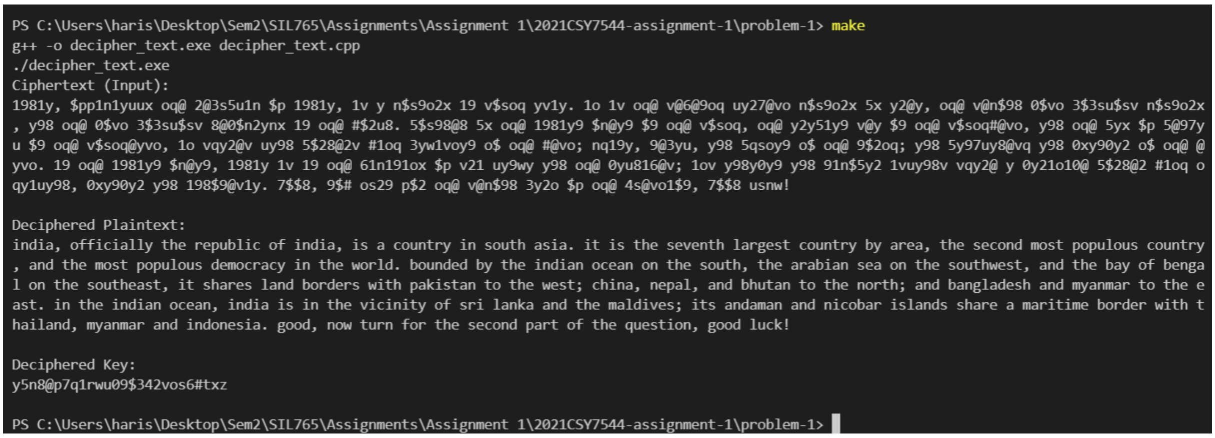
Deciphered Plaintext:

defeated and leaving his dinner untouched, he went to bed. that night he did not sleep well, having feverish dreams, having no rest. he was unsure whether he was asleep or dreaming. conscious, unconscious, all was a blur. he remembered crying, wishing, hoping, begging, even laughing. he floated through the universe, seeing stars, planets, seeing earth, all but himself. when he looked down, trying to see his body, there was nothing. it was just that he was there, but he could not feel anything for just his presence.

Deciphered Key:

8ot64spnrxzqwy$173vu2059#@

Outputs



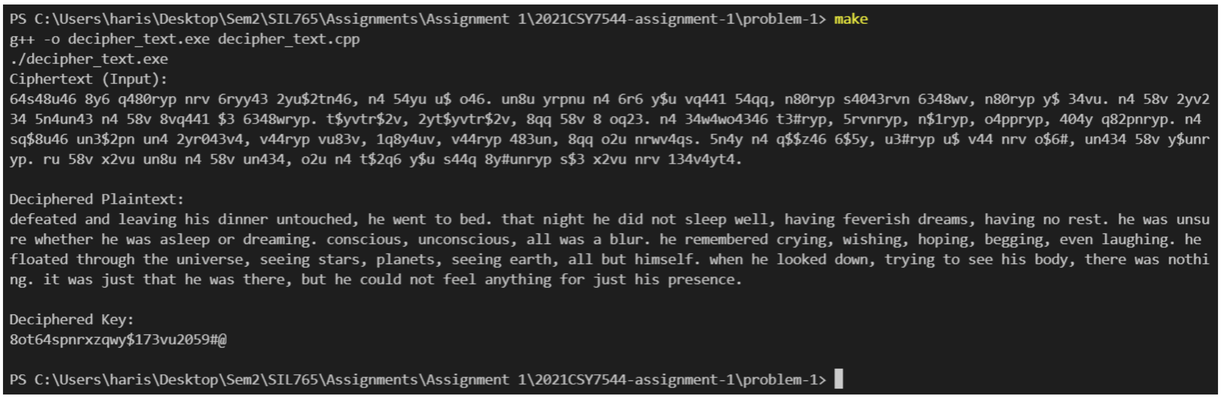
```
PS C:\Users\haris\Desktop\Sem2\SIL765\Assignments\Assignment 1\2021CSY7544-assignment-1\problem-1> make
g++ -o decipher_text.exe decipher_text.cpp
./decipher_text.exe
Ciphertext (Input):
1981y, $pp1n1yuux oq@ 2@3s5u1n $p 1981y, 1v y n$s9o2x 19 v$soq yv1y. 1o 1v oq@ v@6@9oq uy27@vo n$s9o2x 5x y2@y, oq@ v@n$98 0$vo 3$3su$sv n$s9o2x
, y98 oq@ 0$vo 3$3su$sv 8@0$n2ynx 19 oq@ #$2u8. 5$s98@8 5x oq@ 1981y9 $n@y9 $9 oq@ v$soq, oq@ y2y51y9 v@y $9 oq@ v$soq#@vo, y98 oq@ 5yx $p 5@97y
u $9 oq@ v$soq@yvo, 1o vqy2@v uy98 5$28@2v #1oq 3yw1voy9 o$ oq@ #@vo; nq19y, 9@3yu, y98 5qsoy9 o$ oq@ 9$2oq; y98 5y97uy8@vq y98 0xy90y2 o$ oq@ @
yvo. 19 oq@ 1981y9 $n@y9, 1981y 1v 19 oq@ 61n191ox $p v21 uy9wy y98 oq@ 0yu816@v; 1ov y98y0y9 y98 91n$5y2 1vuy98v vqy2@ y 0y21o10@ 5$28@2 #1oq o
qy1uy98, 0xy90y2 y98 198$9@v1y. 7$$8, 9$# os29 p$2 oq@ v@n$98 3y2o $p oq@ 4s@vo1$9, 7$$8 usnw!

Deciphered Plaintext:
india, officially the republic of india, is a country in south asia. it is the seventh largest country by area, the second most populous country
, and the most populous democracy in the world. bounded by the indian ocean on the south, the arabian sea on the southwest, and the bay of benga
l on the southeast, it shares land borders with pakistan to the west; china, nepal, and bhutan to the north; and bangladesh and myanmar to the e
ast. in the indian ocean, india is in the vicinity of sri lanka and the maldives; its andaman and nicobar islands share a maritime border with t
hailand, myanmar and indonesia. good, now turn for the second part of the question, good luck!

Deciphered Key:
y5n8@p7q1rwu09$342vos6#txz

PS C:\Users\haris\Desktop\Sem2\SIL765\Assignments\Assignment 1\2021CSY7544-assignment-1\problem-1> █
```

Fig.1. Output for Ciphertext 1



```
PS C:\Users\haris\Desktop\Sem2\SIL765\Assignments\Assignment 1\2021CSY7544-assignment-1\problem-1> make
g++ -o decipher_text.exe decipher_text.cpp
./decipher_text.exe
Ciphertext (Input):
64s48u46 8y6 q480ryp nrv 6ryy43 2yu$2tn46, n4 54yu u$ o46. un8u yrpnu n4 6r6 y$u vq441 54qq, n80ryp s4043rvn 6348wv, n80ryp y$ 34vu. n4 58v 2yv2
34 5n4un43 n4 58v 8vq441 $3 6348wryp. t$yvtr$2v, 2yt$yvtr$2v, 8qq 58v 8 oq23. n4 34w4wo4346 t3#ryp, 5rvnryp, n$1ryp, o4ppryp, 404y q82pnryp. n4
sq$8u46 un3$2pn un4 2yr043v4, v44ryp vu83v, 1q8y4uv, v44ryp 483un, 8qq o2u nrwv4qs. 5n4y n4 q$$z46 6$5y, u3#ryp u$ v44 nrv o$6#, un434 58v y$unr
yp. ru 58v x2vu un8u n4 58v un434, o2u n4 t$2q6 y$u s44q 8y#unryp s$3 x2vu nrv 134v4yt4.

Deciphered Plaintext:
defeated and leaving his dinner untouched, he went to bed. that night he did not sleep well, having feverish dreams, having no rest. he was unsu
re whether he was asleep or dreaming. conscious, unconscious, all was a blur. he remembered crying, wishing, hoping, begging, even laughing. he
floated through the universe, seeing stars, planets, seeing earth, all but himself. when he looked down, trying to see his body, there was nothi
ng. it was just that he was there, but he could not feel anything for just his presence.

Deciphered Key:
8ot64spnrxzqwy$173vu2059#@

PS C:\Users\haris\Desktop\Sem2\SIL765\Assignments\Assignment 1\2021CSY7544-assignment-1\problem-1> █
```

Fig.2. Output for Ciphertext 2

\