CIS 3362 Homework #2: Substitution Cipher, Vigenere Solutions
By Daniel Gordon

Part A: Code Break Questions

1) Decode the following message, which was encrypted using the substitution cipher. Make sure to discuss all the steps you took, the key you arrived at, and the decoded message.

```
ymlwrpnndvyxijlriewyficlwqvyjvlrislwtpmxsvlihiwyjcywjqqlzwip
tqvirijjpuiopmuiqyqysynnxlqvijpriqvymuqvyjdipwamclwqampqindq
vyjyjmqqvirijjpuiqvpqqinnjdlasviwiqviewyfiyjwpqviwqviblzlcqv
yjrijjpuiyjqlxijowyziqvpqqviwiyjlmibajqnytiqvizipnioyeviwzaq
amnytizipniqvyjewyfisynnpoqapnndigyjq
```

Solution

Cryptool.html was used for this question, as well as the formula sheet.
Pasting in the ciphertext, we get the following frequency breakdown:

A	2.5%	N	5.4%
B	0.7%	O	1.4%
C	1.4%	P	6.1%
D	1.8%	Q	11.2%
E	1.4%	R	2.5%
F	1.1%	S	1.8%
G	0.4%	T	1.4%
H	0.4%	U	1.8%
I	**15.2%**	V	7.6%
J	8.3%	W	6.1%
K	0%	X	1.4%
L	5.1%	Y	9.4%
M	3.2%	Z	2.2%

Frequency Information of each letter in regular English text

Let	A	B	C	D	E	F	G	H	I	J	K	L	M
Frq	.082	.015	.028	.043	.127	.022	.020	.061	.070	.002	.008	.040	.024
Let	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frq	.067	.075	.019	.001	.060	.063	.091	.028	.010	.023	.001	.020	.001

Table from the reference sheet

Ciphertext

I	15.2%	U	1.8%
Q	11.2%	D	1.8%
Y	9.4%	S	1.8%
J	8.3%	E	1.4%
V	7.6%	C	1.4%
W	6.1%	X	1.4%
P	6.1%	O	1.4%
N	5.4%	T	1.4%
L	5.1%	F	1.1%
M	3.2%	B	0.7%
A	2.5%	G	0.4%
R	2.5%	H	0.4%
Z	2.2%	K	0%

English Average

E	12.7%	M	2.4%
T	9.1%	W	2.3%
A	8.2%	F	2.2%
O	7.5%	Y	2.0%
I	7.0%	G	2.0%
N	6.7%	P	1.9%
S	6.3%	B	1.5%
H	6.1%	V	1.0%
R	6.0%	K	0.8%
D	4.3%	J	0.2%
L	4.0%	Q	0.1%
U	2.8%	X	0.1%
C	2.8%	Z	0.1%

We see here that I and Q are the two most frequently used letters. If we map these to E and T, we get the following.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------E-------T---------
```

Decipher

```
------------E---E----E---T------E----------E-E-------TT---E-
-T-E-E----E----ET-T-------T-E---ET----T----E-------T---TE--T
------TT-E-E----ET--TTE--------E-ET-E----E----T-E-T-E-----T-
---E----E--T--E-----ET--TT-E-E----E---T---ET-E-E--E----E---T
-----E-E--ET-------E------T-----E---T
```

Now that we have a probable T and E, we can look for a trigram that could be "THE".

Find Repeated N-grams

```
PNND XIJ LRI IEWYFI CLWQ WQV QVY QVYJ QVI
QVIRIJJPUI UIQ SYNN IQV IQVY QAM QIN QQVI
IQVPQQ VIW VIWI IQVI IYJ QVIW YJQ NYTI IZIPNI
```

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------E-------T---------
```

QVI stands out as a probable trigram, so let's try it.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------E-------T----H----
```

Decipher

```
---------H--E---E----E---TH--H--E--------H-E-E-------TT---E-
-THE-E----E----ET-T-------THE---ETH---TH---E-------T---TE--T
H-----TTHE-E----ETH-TTE-------HE-ETHE----E----THE-THE-----TH
---E----E--T--E-----ETH-TTHE-E----E---T---ETHE-E--E---HE---T
-----E-E--ETH------E------T-----E---T
```

There are 2 `TH-T` patterns, which could be `THAT`.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------E------AT----H----
```

Decipher

```
-----A---H--E---E----E---TH--H--E----A---H-E-E-------TT---EA
-THE-E--A-E-A--ET-T-------THE-A-ETH---TH---EA------T--ATE--T
H-----TTHE-E--A-ETHATTE-------HE-ETHE----E---ATHE-THE-----TH
---E--A-E--T--E-----ETHATTHE-E----E---T---ETHE-EA-E---HE---T
-----E-EA-ETH------E----A-T-A---E---T
```

Next, we'll look for `THIS` in the N-grams, which should be `QV??`.

Find Repeated N-grams

```
PNND XIJ LRI IEWYFI CLWQ WQV QVY QVYJ QVI
QVIRIJJPUI UIQ SYNN IQV IQVY QAM QIN QQVI
IQVPQQ VIW VIWI IQVI IYJ QVIW YJQ NYTI IZIPNI
```

`QVYJ` looks the part, so let's try it.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES-----AT----H--I-
```

Decipher

```
I----A---HI-ES--E--I-E---THISH--E----A---H-E-E-IS-I-STT---EA
-THE-ESSA-E-A--ETITI-I----THESA-ETHI--THIS-EA------T--ATE--T
HISIS-TTHE-ESSA-ETHATTE--S----HE-ETHE--I-EIS-ATHE-THE-----TH
IS-ESSA-EIST--ES--I-ETHATTHE-EIS--E--ST-I-ETHE-EA-E-I-HE---T
---I-E-EA-ETHIS--I-E-I--A-T-A---E-IST
```

`SA-E` appears a few times, and 3 of them are the same pattern. It might map to `SAME`.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES-----AT---MH--I-
```

Decipher

```
I----A---HI-ES--E--I-E--- THIS H--E----A---H-E-E-IS-I-STT---EA
- THE -ESSAME-A-METITI-I---- THE SA-ETHI-M THIS -EA------T--ATE--T
HISIS-T THE -ESSAMETHATTE--S----HE-ETHE--I-EIS-ATHE-THE-----TH
IS-ESSAMEIST--ES--I-E THAT THE -EIS--E--ST-I-ETHE-EA-E-I-HE---T
---I-E-EA-E THIS --I-E-I--A-T-A---E-IST
```

Using the `SA-E` pattern that appears 3 times doesn't look quite right. Let's try the other one.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES-----ATM---H--I-
```

Decipher

```
I---MA---HI-ES-ME--I-E--- THIS H-ME----A---H-E-E-IS-I-STT---EA
- THE MESSA-E-A--ETITI-I---- THE SAMETHI-- THIS -EA------T--ATE--T
HISIS-T THE MESSA-ETHATTE--S----HE-ETHE--I-EIS-ATHE-THE-----TH
ISMESSA-EIST--ES--I-E THAT THE -EIS--E--ST-I-ETHE-EA-E-I-HE---T
---I-E-EA-E THIS --I-E-I--A-T-A---E-IST
```

This looks a lot better! And the pattern `MESSA-E` appears three times, which could be `MESSAGE`.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES-----ATM--GH--I-
```

Decipher

```
I---MA---HI-ES-ME--I-E--- THIS H-ME----A---H-E-E-IS-I-STT---EA-
THE MESSAGE -A-GETITI-I---- THE SAME THI-G THIS -EA------T--ATE--
THIS IS-T THE MESSAGE THAT TE--S----HE-ETHE--I-EIS-ATHE-THE-----
THIS MESSAGE IST--ES--I-E THAT THE -EIS--E--ST-I-ETHE-EA-E-I-HE---T
---I-E-EA-E THIS --I-E-I--A-T-A---E-IST
```

`THI-G` is probably `THING`.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES--N--ATM--GH--I-
```

Decipher

```
IN--MA---HI-ES-ME--I-E--- THIS H-ME----AN--H-E-E-IS-I-STT---EA-
THE MESSAGE -ANGETITI-I---- THE SAME THING THIS -EA--N---T-NATE--
THIS ISNT THE MESSAGE THAT TE--S----HE-ETHE--I-EIS-ATHE-THE-----
THIS MESSAGE IST--ES--I-E THAT THE -EIS-NE--ST-I-ETHE-EA-E-I-HE---T
-N-I-E-EA-E THIS --I-E-I--A-T-A---E-IST
```

There are two `THE-E` patterns, one of which might be `THERE`. (`QVIWI` and `QVIZI`)

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES--N--ATM--GH--I-
```

Decipher

```
IN--MA---HI-ES-ME--I-E--- THIS H-ME----AN--H-E-E-IS-I-STT---EA-
THE MESSAGE -ANGETITI-I---- THE SAME THING THIS -EA--N---T-NATE--
THIS ISNT THE MESSAGE THAT TE--S----HE-ETHE--I-EIS-ATHE-THE-----
THIS MESSAGE IST--ES--I-E THAT THE-EIS-NE--ST-I-ETHE-EA-E-I-HE---T
-N-I-E-EA-E THIS --I-E-I--A-T-A---E-IST
```

With `QVIWI`:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES--N--ATM--GHR-I-
```

Decipher

```
IN-RMA---HI-ES-ME-RI-E--R THIS H-ME--R-AN--H-E-ERIS-IRSTT--REA-
THE MESSAGE -ANGETITI-I---- THE SAME THING THIS -EAR-N--RT-NATE--
THIS ISNT THE MESSAGE THAT TE--S----HERETHE-RI-EISRATHERTHE-----
THIS MESSAGE IST--ES-RI-E THAT THEREIS-NE--ST-I-ETHE-EA-E-I-HER--T
-N-I-E-EA-E THIS -RI-E-I--A-T-A---E-IST
```

With `QVIZI`:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES--N--ATM--GH--IR
```

Decipher

```
IN--MA---HI-ES-ME--I-E--- THIS H-ME----AN--H-E-E-IS-I-STT-R-EA-
THE MESSAGE -ANGETITI-I---- THE SAME THING THIS -EA--N---T-NATE--
THIS ISNT THE MESSAGE THAT TE--S----HE-ETHE--I-EIS-ATHE-THE--R--
THIS MESSAGE IST--ES--IRE THAT THE-EIS-NE--ST-I-ETHEREA-E-I-HE-R-T
-N-I-EREA-E THIS --I-E-I--A-T-A---E-IST
```

The QVIWI pattern gives us a new word, RATHER, so it appears that it is correct!

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES--N--ATM--GHR-I-
Decipher
IN-RMA---HI-ES-ME-RI-E--R THIS H-ME--R-AN--H-E-ERIS-IRSTT--REA-
THE MESSAGE -ANGETITI-I---- THE SAME THING THIS -EAR-N--RT-NATE--
THIS ISNT THE MESSAGE THAT TE--S----HERETHE-RI-EIS RATHER THE-----
THIS MESSAGE IST--ES-RI-E THAT THEREIS-NE--ST-I-ETHE-EA-E-I-HER--T
-N-I-E-EA-E THIS -RI-E-I--A-T-A---E-IST
```

Moving back to common words, there's an AN- that might be AND.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES--N--ATM--GHRDI-
Decipher
IN-RMA--- HIDE S-ME-RI-E--R THIS H-ME--R- AND -H-E-ERIS-IRSTT--REA-
THE MESSAGE -ANGETITI-I--D- THE SAME THING THIS -EAR-N--RT-NATE--
THIS ISNT THE MESSAGE THAT TE--S----HERETHE-RI-EIS RATHER THE-----
THIS MESSAGE IST-DES-RI-E THAT THERE IS -NE--ST-I-ETHE-EA-E-I-HER--T
-N-I-E-EA-E THIS -RI-E-I--A-T-A---E-IST
```

S-ME might be SOME.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES-ON--ATM--GHRDI-
Decipher
I NORMA--- HIDE SOME -RI-E-OR THIS HOME-OR- AND -HOE-ERIS-IRSTTO-REA-
THE MESSAGE -ANGETITI-I--DO THE SAME THING THIS -EAR-N-ORT-NATE--
THIS ISNT THE MESSAGE THAT TE--S-O-- HERE THE-RI-EIS RATHER THE-O-O-
THIS MESSAGE IS TO DES-RI-E THAT THERE IS ONE--ST-I-ETHE-E
A-E-I-HER--T -N-I-E-EA-E THIS -RI-E-I--A-T-A---E-IST
```

NORMA- is looking quite close to NORMAL.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
--------ES-ONL-ATM--GHRDI-
```

Decipher

```
I NORMALL- HIDE SOME -RI-E-OR THIS HOME-OR- AND -HOE-ERIS-IRSTTO-REA-
THE MESSAGE -ANGETITI-ILL DO THE SAME THING THIS -EAR-N-ORT-NATEL-
THIS ISNT THE MESSAGE THAT TELLS -O-- HERE THE-RI-EIS RATHER THE-O-O-
THIS MESSAGE IS TO DES-RI-E THAT THERE IS ONE--STLI-ETHE-E
ALE-I-HER--T -NLI-E-EALE THIS -RI-E-ILLA-T-ALL-E-IST
```

And NORMALL- is probably NORMALLY

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
---Y----ES-ONL-ATM--GHRDI-
```

Decipher

```
I NORMALLY HIDE SOME -RI-E-OR THIS HOME-OR- AND -HOE-ERIS-IRSTTO-REA-
THE MESSAGE -ANGETITI-ILL DO THE SAME THING THIS YEAR -N-ORT-NATELY
THIS ISNT THE MESSAGE THAT TELLS YO-- HERE THE-RI-EIS RATHER THE-O-O-
THIS MESSAGE IS TO DES-RI-E THAT THERE IS ONE--STLI-ETHE-E
ALE-I-HER--T -NLI-E-EALE THIS -RI-E-ILLA-T-ALLYE-IST
```

-ORT-NATELY could be FORTUNATELY.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
U-FY----ES-ONL-ATM--GHRDI-
```

Decipher

```
I NORMALLY HIDE SOME -RI-E FOR THIS HOME-OR- AND -HOE-ER IS FIRST TO-REA-
THE MESSAGE -AN GET IT I- ILL DO THE SAME THING THIS YEAR UNFORTUNATELY
THIS ISNT THE MESSAGE THAT TELLS YOU -HERE THE -RI-EIS RATHER THE -O-OF
THIS MESSAGE IS TO DES-RI-E THAT THERE IS ONE-USTLI-ETHE-E
ALE-I-HER-UTUNLI-E-EALE THIS -RI-E-ILLA-TUALLYE-IST
```

HOME-OR- could be HOMEWORK.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
U-FY----ES-ONL-ATMWKGHRDI-
```

Decipher

```
I NORMALLY HIDE SOME -RI-E FOR THIS HOMEWORK AND WHOE-ER IS FIRST TO -REAK
THE MESSAGE -AN GET IT I WILL DO THE SAME THING THIS YEAR UNFORTUNATELY
THIS ISNT THE MESSAGE THAT TELLS YOU WHERE THE -RI-E IS RATHER THE -O-OF
THIS MESSAGE IS TO DES-RI-E THAT THERE IS ONE -UST LIKE THE -EALE
-I-HER-UT UNLIKE -EALE THIS -RI-E WILL A-TUALLY E-IST
```

And WHOE-ER is probably WHOEVER, while -REAK looks a lot like BREAK.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
U-FY---VES-ONL-ATMWKGHRDIB
Decipher
I NORMALLY HIDE SOME -RI-E FOR THIS HOMEWORK AND WHOEVER IS FIRST TO BREAK
THE MESSAGE -AN GET IT I WILL DO THE SAME THING THIS YEAR UNFORTUNATELY
THIS ISNT THE MESSAGE THAT TELLS YOU WHERE THE -RI-E IS RATHER THE -OBOF
THIS MESSAGE IS TO DES-RIBE THAT THERE IS ONE -UST LIKE THE BEALE
-I-HERBUT UNLIKE BEALE THIS -RI-E WILL A-TUALLY E-IST
```

DES-RIBE maps to DESCRIBE, while E-IST looks a lot like EXIST.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
U-FY--XVES-ONLCATMWKGHRDIB
Decipher
I NORMALLY HIDE SOME -RI-E FOR THIS HOMEWORK AND WHOEVER IS FIRST TO BREAK
THE MESSAGE CAN GET IT I WILL DO THE SAME THING THIS YEAR UNFORTUNATELY
THIS ISNT THE MESSAGE THAT TELLS YOU WHERE THE -RI-E IS RATHER THE -OB OF
THIS MESSAGE IS TO DESCRIBE THAT THERE IS ONE -UST LIKE THE BEALE
CI-HER BUT UNLIKE BEALE THIS -RI-E WILL ACTUALLY EXIST
```

-OB is probably JOB, while C-PHER is CIPHER.

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
UJFYP-XVES-ONLCATMWKGHRDIB
Decipher
I NORMALLY HIDE SOME PRI-E FOR THIS HOMEWORK AND WHOEVER IS FIRST TO BREAK
THE MESSAGE CAN GET IT I WILL DO THE SAME THING THIS YEAR UNFORTUNATELY
THIS ISNT THE MESSAGE THAT TELLS YOU WHERE THE PRI-E IS RATHER THE JOB OF
THIS MESSAGE IS TO DESCRIBE THAT THERE IS ONE JUST LIKE THE BEALE
CIPHER BUT UNLIKE BEALE THIS PRI-E WILL ACTUALLY EXIST
```

And finally, the end is in sight! PRI-E is PRIZE, and the only other unused letter is Q, so we can just slap that in the last slot via process of elimination. There we have it!

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
UJFYPZXVESQONLCATMWKGHRDIB
Decipher
I NORMALLY HIDE SOME PRIZE FOR THIS HOMEWORK AND WHOEVER IS FIRST TO BREAK
THE MESSAGE CAN GET IT I WILL DO THE SAME THING THIS YEAR UNFORTUNATELY
THIS ISNT THE MESSAGE THAT TELLS YOU WHERE THE PRIZE IS RATHER THE JOB OF
THIS MESSAGE IS TO DESCRIBE THAT THERE IS ONE JUST LIKE THE BEALE
CIPHER BUT UNLIKE BEALE THIS PRIZE WILL ACTUALLY EXIST
```

2) Decode the following ciphertext that was encoded using the Vigenere cipher with a keyword from this wordlist:

https://www.ef.com/wwen/english-resources/english-vocabulary/top-1000-words/

To help you automate your task, I will guarantee that the substring "last" will appear somewhere in the plaintext. Here is the ciphertext:

```
sxvgjcvpsamltorbqmwaskvytneeyawceflrirtdrwmtoheiutlqxuzbzarbi
bilphbiblgrflrfayhvoacwxqkvoiyksmmvabzilmodszzkfacmrllotrfomr
ruqvrbsssjrmnwyxhbbtelvnbzaruaveldxvgtywqqwbfeelviorbqflnmszq
ropzbtuwngoeijcqtvgqpylyzyjcmisgakbmorignwdspmwuloelrgijhzqwf
tgpuhzakwbzxukepavgtyoiqxuxcwyrowwwapeaxngiyjxvkuugupiwptddvm
bghvkenxvjvjhbopnbmjshmzvkndhthoophxs
```

Solution

To solve this, I copied the word list into words.txt, and created a program to loop through each possible keyword. The program is attached.

The most important line of the code was the following, which looped through the ciphertext and the key and added them together:

```
for(int j = 0; j < strlen(ciphertext); j++)
{
    printf("%c", 'a' + ((ciphertext[j] + 26 - 'a' - (key[j % strlen(key)] - 'a')) % 26));
}
```

The program produced the following output when run:

```
hallowed@Hallowed:~/CIS3362TA$ gcc HW2Q2.c -o hw2q2.exe && ./hw2q2.exe


======== QUESTION 2 ========

Key          a: sxvgjcvpsamltorbqmwaskvytneeyawceflrirtdrwmtoheiutlqxuzbzarbibilphbiblgrflrfayhvoacwxqkvoiyksmmvabzilmods
zzkfacmrllotrfomrruqvrbsssjrmnwyxhbbtelvnbzaruaveldxvgtywqqwbfeelviorbqflnmszqropzbtuwngoeijcqtvgqpylyzyjcmisgakbmorignwd
spmwuloelrgijhzqwftgpuhzakwbzxukepavgtyoiqxuxcwyrowwwapeaxngiyjxvkuugupiwptddvmbghvkenxvjvjhbopnbmjshmzvkndhthoophxs
Key    ability: swnvbjxprsbdaqraibohukuqiflgyzorwmnrhjivyymsgwwpwtkimmgdzzjqaiklozqaingqxajmcygndsjyxpckgpakrebnhdzhdbgku
zycusjorkddlyholjgmxxrakhkqtmmonpodbswanudzzjjscglcpkyaampiltmgeknxgydqedcezbqqgerivuvfvglkjbiinnspxdnrflclahyhmblgganpwc
keedwlnwajnkjgrfomvgomwrhmwarmmrgpznvlfqippjpjyyqglodcpdsmfnkyipkcbmgthxowvdcnbtnjvjwcpclvizqgwpblbhztbvjfszajonhwpz
Key       able: swkcjbklszbhtngxqllwsjkutmtayzlyeeaniqizrvbpogteusamxtoxzzgxiaxhpgqebkvnfkgbaxwrozrsxpzrohngslbraaoelldzs
yogfzrirkaktqukmqgqqugxsrhfrlcsywwxbsthvmqvaqjwvdazxuvpyvfmwauaekkeoqqmfkcisyfnooxttljgntejbfpvfflyknvyiriirvwkabkrhvjwc
hlmvjhodanghydzplbtfeqhypgwaotujtlauvpynxmxtmywxgkwvlwpdptnfxujwkgutvqphlltcsrmavdvjtjxuyrjgqkpmqijrwizuzjdgidonedxr
Key      about: swhmqcubyhmkfuybpychsjheandqehwbqlsrhdzkrvyzvhduaalpjagbymxiiaurwhauhsgqrryfzkncozoceqjhupyjestvznfpllajz
zywlhcldrsosdlvmqdaxvqnyzsidsuwxjnibsqrcnalgyuzhksdwhmayvcwdbeqksvhaxiqexttsycxvpynzbwmsuliiowavfcvflxleqcluynajnsvrhstdd
rbsdukaksrfupozpilagognqajihgxtwkwauszfohcdbxbieyovichpdmdughkpevjganuoucwtcpbtbftbremjbqvithvpmnsqsgyfckmpnahnavoxr
Key      above: swhlfcubxwmkftnbpybwsjhdpndqdwwbqkhrhdyzrvyykhduzplpjzvbymwxiauqlhaughgqrqnfzkmrozobtqjhteyjerivzneellaio
zywkwcldqhosdkkmqdzmvqnxosidrjwxjmxbsqqrnalfnuzhjhdwhlpyvcvsbeqjhvhamxqexsisycwkpynyqwmstaiiovpvfcuulxldfcluxcajnrkrhsssd
rbrsukajhrfuodzpikpgogmvajigvxtwjlausyuohccqxbidnovibwpdmcjghkotvjgzcuoubltcpaibftagemjafvitgkpmnrfsgyerkmpmphnaudxr
Key     accept: svtcujvnqwxstmpxbtwyqggftlcajhwacbwyiprzcdmrmdppurjmibzzxwciizghaobgzhryfjpblfhtmwndxoirzpyiqixcazxemtobq
vkrfyaicslmrnqvmppqbcrzqodqrklsjehzzppsvlzvlyuytawkxtepjdqouxqlejtezybodhytsxonzwzzrqhugmceujqrtcbwyjwvjqckgorhkzkkcpgluz
dwmushzllpeeuozoubenpsfvlrwzxtfrenyrraymgmibxauucvwuuwalavlctfjvtgfbgsnehwtbbrxigftgpuxthruobmnjmtjqfikcklbdeoomndiz
Key  according: svtssznemakjfxotdgmyqwevlayemyilbxylipzpategihcggciikozzxmayaoclnfnrydtlfjprjvziiaaujzhnbcyiqyvssotijkamp
rmefyayaidbnrdmyaomdprzqebgjzhwwvtkylrfvlzljomnpejbjedllqqounobwypimpnzcdagsxodxmronuulsxbamwqrtszmqyszmhovfktukzkaafyaqd
qnyfrdbylpeuserdqfrebdernewzxjdhwcuverkxfikoxaukalojqancmgkyvsjvtwdryhjiunfmanzvgftwnkpidvhfnxmfogjqfyiscaxhrfaxmzkm
Key    account: svtsppcpqyyrgvrzoycnzktwftrlyyuokssrgpfjedmrmtkvbtjojamizypnooplnfnoosgpdxxshyftagpdxoihuvfkqkybnizgjyuqz
zxirgptrjjazemokpdadcrzqeywymlukduibrcxbaizypggillbvhmgfwooihslejtuueiqdjzsfgqpmbfoauulsurpjaofbtxpwjkflqckgemnrbkmdotuwb
qbsjblmcxxtpjfxccsagnstfnrwxjaxlpytszlviovgdpdypmicjhpcyjttpyhvhqhbgsnuccadbtyhtoviczdiqvhfnucubkhenzgvilpngoomntdf
Key     access: svesrkxnbmuttpanyuuvbwdotlpqpimmortzipcnyzmrxtmquuvcfczzimziizrxxobokroxfiorjqhtxwkexothmnvihvudazjutuohb
```

To find the correct key, I searched the output for the word “last”:

```
hallowed@Hallowed:~/CIS3362TA$ gcc HW2Q2.c -o hw2q2.exe && ./hw2q2.exe | grep "last"
Key environmental: okaysoidontlikegivinggiftcardsfortheprizebecausehalfthetimepeopledontusethemanditslikegivingfreemoneyt
osomecompanysoinsteadhereishowwewilldoitthistimeiwillstillhidesomethingandthatsomethingisanotewhichtellsyouwheretogotoget
somecashsothelastmessagequestionthreewillhavethecluetofindanenvelopewhichwillgiveyoudirectionstoclaimyourrewardgoodluck
Key       last: hxdnycdwhausiozifmehhkdfinmlnaejtftyxrbkgmuadhmpjttxmuhioazixbqsehjpqloyulzmpypcdakdmqscdigrhmucpbhpamwkh
zhruaktgltvirnvbrzbfvzihsaqgmvdnxpiqtmsknjgprchketkmvoanwyxlbnltldpdrjxulvthzyydphiiueuvompycyakgywnlggnjktxsohzbuvgiould
awbwcsdetyvirooqemigxbwzirlbhejkmwpvoanoqxmufjlyzvlweheeiecgqfyxdrjuobeiewidlcbbookkmumvrcyhjvenjtysptovsushbodoxoms
Key  religious: btkyduhvajialijnwufwhcpqftmnupowmrrzrnivloyzwqaxmndcdcixosltuhqulwtctxmzohgxuqtbwjylpkchuqhghegnmhhrhbgxk
lfsowreldxubabdeljgmdaxhmbdsvfumzvtfktejqrujggdnhsppyfeezmltzwqrdrkgtkxxtubvfjihlhbdscyimupkzpkykhkrgiuyugaemitxbglastem
oeeqmxumunvadzlweopvhozlgsfxopocqvieciqiacdcgylqlgicejltsrfsogstkcomsaxrselxvhsjpdkcyfjbrefwtihzhusometnwtlqpwgihtda
Key       site: apccruclasthbgyxyedwaccubflagsdymxsnqjazzotpwzleclsmfwgxhsyxqtphxziejdnnndybiqorwsjsfirrwafgaetritgetevza
rggnsjizdskbjmkujyqynyxakzfzeusgpoxjllhdfivijbwdwszfnnpgoxmetmamdcewjimnduiarxnwhgxbmdjogleruxpdyxlgdfvgbjiqknwsttkzanjev
zluobhwwsnoaqdhidbbywqprhgetgtcellinnpgggmfmeyeqykeodwxwhtvypurpcgcmnqxadlbvkrutnddcljfnqrrzikxfiirkoihnrjlzadwgwdfk
```

I got 4 lines, but only the first one actually made sense. The key was "`environmental`" and the plain text was

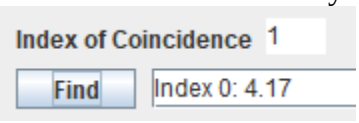
```
Okay so I don't like giving gift cards for the prize because
half the time people don't use them and it's like giving free
money to some company so instead here is how we will do it this
time I will still hide something and that something is a note
which tells you where to go to get some cash so the last message
question three will have the clue to find an envelope which will
give you directions to claim your reward good luck
```

3) Decode the following message, which was encrypted using the Vigenere cipher. Make sure to discuss all the steps you took, the key you arrived at, and the decoded message.

```
jiaikmiwriuopleicwmrdeuiypieepcefhsdewbcssfzrtlraahfjsnyunyel
enusbxvdnwkplikgkccxpmdwreetsffbeosplaemhyzzilddmszqhbvzolous
uecsxkafaxzahzavqzxllohpyfskbvdelycstfazuzytzoniwrprnymtkbael
cwofcfhaciisjyuyeqevcpssrwpokfoekgiikcomzqrxyncolrcwimccswqnf
ufudnlaxcwokfonpauzkrimfpswymrsmpirjrcojqfgbaecyzwhrzdqyqvvrp
wiletwcpwajgklzfeeyniikgggfdedsgilprvuketsdewtzaofcktzoegaeba
ctfekdoaicjvycbygenicftslktecoumbvrhbvweqgeplemhmldpjsoisfksi
wkom
```

Solution

First, I needed to find the key length, so I used Cryptool.java to calculate the index of coincidence of various key lengths.



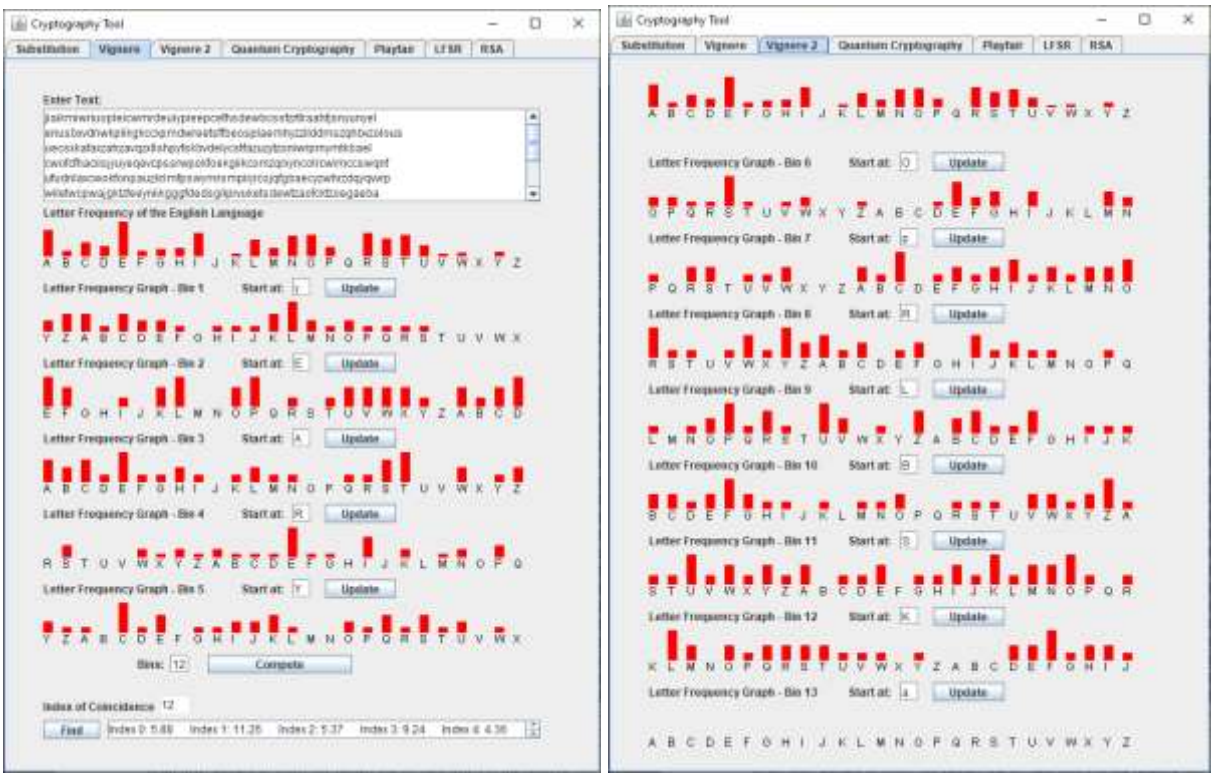
However, the key length with the highest IoCs was 12:

From this, I concluded that the key length was 12.

After that, I began to play around with keys in cryptool.java, and tested them with a program I adapted from question 2 to take a key from user input and decode the ciphertext with it. The first one that seemed valid from a frequency perspective started with `HAPPY`, but when I tried that in my decryption program, it gave me nothing.

```
key>> happyoyrmnqt
Key happyoyrmnqt: ciltmykffvevilpteioarreprptpgbentuckxwmnuehifgvytasqlephiaileeyfunzeragriltviwellcwkprppvehopryzillpota
invvkwmdkstdenbvvnsfpeeztoskesaskchsilyvvapjquwderrvfvseqclwimgjvgihcrdphaguitemnyahltukjbiduaganerfjisdcybqttborzitveaoi
eehfgczwtoyrapmzpqyqwrwmbykevwzvhapyohjrkixqreyhaectiicutoqsesqitenjbijanqafjvgcrikusggjiwluiwnitrofgitvisiorrnzziwathwts
gckxwekcahlygjvxglpdnectrukhatnlhalplqlginqventhrmvnmmgttdekranxpwpotourctzhidqmekfybw
```

I continued to tweak it.




```
Key yearyoprlbsk: learmytfghcerhereixasdcyalingbnnugatgsblueqigsthcwholeyhjmgunanduniesmearhitiwnlmoutynenveqoqdwirhanotj
iohttfisistmeontemouneeitpeinbwhichbimktejlyoummesdtoeotoclfinshepewardyhbssrcallyaqlugiskessagjnfddsrosaybztunmaieiteaxi
fqfopyoutohrbbkiymnowrfmckinesothayypthatemorehhbqacrerstozsfeorcachbisaocyosrvaritutsesrsasiwwiudmopeitisrosdliielythftt
satgsticaqlzshegcandmncudstqmillhjlqxoupecovemtidkewibettmeldywgllnotxusoriqesometfznu
```

That looked almost legible!

```
Key?:           yearyoprlbskyearyoprlbsk
Text:           learmytfghcerhereixasdcy
Possible text: nearmy??????thereisa????
```

So if we adjust the key so the first translated letter is "n" (`y -> w`) and the "x" is an "s" (`p -> u`) we get a key of "`wearyourlbsk`".

Putting that in, we get the first bit as:
`nearmyofghcethereisasdcyclingbinugatisblueligsthewholethjmgupand`
From here, we can make some guesses as to what to change

```
Key?:           wearyourlbskwearyourlbskwearyourlbskwe
Text:           nearmyofghcethereisasdcyclingbinugatis
Possible text: nearmyofficethereisarecyclingbinthatis
```

If we change the key to reflect that, we get a key of "`wearyourmask`" and a plaintext of

```
Near my office there is a recycling bin that is blue. Lift the
whole thing up and underneath it I will put an envelope with a
note in it. This is the note you need to find which will tell
```

you where to go to claim the reward. That’s really all this message needs to say, but to make it easier for you to crack I am now rambling so that you have more characters to use for each bin and your various tests. As I write more, it is more likely that statistically the IC and MIC tests will help you recover the key. I bet the key will not surprise some of you.

Part B: Written Questions Similar to Quiz/Exam Questions

4) Find 45^{-1} mod 157

Solution

For this, we'll use the EEA.

```
157 = 45 * 3 + 22   => 22 = 157 – (3 * 45)
 45 = 22 * 2 + 1    =>  1 =  45 – (2 * 22)

1 = 45 – (2 * 22)
1 = 45 – (2 * (157 – (45 * 3))
1 = 45 – ((2 * 157) – (6 * 45))
1 = 45 – (2 * 157) + (6 * 45)
1 = (7 * 45) – (2 * 157)
(7 * 45) = 1 mod 157
```

45^{-1} mod 157 is 7.

5) For an alphabet of size 93, a set of affine encryption keys is a = 20, b = 87. (Thus the encryption function is f(x) = (20x + 87) % 93.) Determine the corresponding set of decryption keys.

Solution

For an Affine encryption formula of $f(x) = (ax + b) \bmod L$ (where L is the length of the alphabet), the decryption key will be found by inverting the encryption formula and solving for x (i.e. $d(x) = c(x - b) \bmod L$, where c is the modular inverse of $a \bmod L$.)

In simpler terms, we need to find the modular inverse of a, which is 20 in this case, for a mod of 93. Then, we'll invert the function, and have the two keys for decryption.

Using the EEA to find $20^{-1} \bmod 93$:

```
93 = 20 * 4 + 13
20 = 13 * 1 +  7
13 =  7 * 1 +  6
 7 =  6 * 1 +  1

1 = 7 - 6
1 = 7 - (13 - 7)
1 = 7 - 13 + 7
1 = -13 + (2 * 7)
1 = -13 + (2 * (20 - 13))
1 = -13 + (2 * 20) - (2 * 13)
1 = (2 * 20) - (3 * 13)
1 = (2 * 20) - (3 * (93 - (20 * 4)))
1 = (2 * 20) - ((3 * 93) - 3(20 * 4))
1 = (2 * 20) - (3 * 93) + (12 * 20)
1 = (14 * 20) - (3 * 93)
14 * 20 = 1 mod 93
20^-1 mod 93 = 14
```

Now, we can begin inverting the function:

$$x = (20y + 87) \bmod 93$$
$$(x - 87) = 20y \bmod 93$$
$$14(x - 87) = (14 * 20)(y) \bmod 93$$
$$14x - 1218 = y \bmod 93$$
$$14x + 84 = y \bmod 93$$
$$d(x) = (14x + 84) \bmod 93$$

Thus, our decryption key is $a = 14$, $b = 84$.

6) Let x be a positive integer. A set of letters consists of 10 As, 30 Bs, 50 Cs, 70 Ds, and 90 Es. What is the index of coincidence of the set? **Leave your answer as a fraction in lowest terms.**

Solution

The size of the set is `10 + 30 + 50 + 70 + 90 = 250` letters.

IOC calculation:

$$\text{Index of coincidence forumla: } \sum \frac{x(x-1)}{n(n-1)}$$

$$= \frac{10(9)}{250(249)} + \frac{30(29)}{250(249)} + \frac{50(49)}{250(249)} + \frac{70(69)}{250(249)} + \frac{90(89)}{250(249)}$$

$$= \frac{10 * 9 + 30 * 29 + 50 * 49 + 70 * 69 + 90 * 89}{250 * 249}$$

$$= \frac{10(9 + 3 * 29 + 5 * 49 + 7 * 69 + 9 * 89)}{10 * 25 * 249}$$

$$= \frac{(9 + 3 * 29 + 5 * 49 + 7 * 69 + 9 * 89)}{25 * 249}$$

$$= \frac{1625}{25 * 249} = \frac{25 * 65}{25 * 249} = \frac{65}{249}$$

The index of coincidence is $\frac{65}{249}$.

7) The set of letters S consists of 5 As, 40 Bs, 35 Cs, 10 Ds, and 10 Es. The set of letters T consists of 15 As, 10 Bs, 35 Cs, 30 Ds and 15 Es. What is the mutual index of coincidence between sets S and T? **Leave your answer as a fraction in lowest terms.**

Solution

Set S is `5 + 40 + 35 + 10 + 10 = 100` letters, and Set T is `15 + 10 + 35 + 30 + 15 = 105` letters. Using the formula for mutual index of coincidence, we get:

$$\text{Mutual index of coincidence forumla: } \sum \frac{x * y}{n * m}$$

$$= \frac{5 * 15 + 40 * 10 + 35 * 35 + 10 * 30 + 10 * 15}{100 * 105}$$

$$= \frac{5(15 + 8 * 10 + 7 * 35 + 2 * 30 + 2 * 15)}{100 * 105}$$

$$= \frac{25(3 + 8 * 2 + 7 * 7 + 2 * 6 + 2 * 3)}{100 * 105}$$

$$= \frac{3 + 8 * 2 + 7 * 7 + 2 * 6 + 2 * 3}{4 * 105} = \frac{86}{4 * 105} = \frac{43}{2 * 105} = \frac{43}{210}$$

The mutual index of coincidence is $\frac{43}{210}$