ASSIGNMENT 3

Breaking Transposition Ciphers

Sukh Atwal
10-11-2020

Contents

Introduction

The objective of this assignment was to design and implement an application that goal was to break a transposition cipher. This would be done through brute forcing using every possible key length.

The language I used was Python due to ease of use and experience with using it along with the sys and math modules. I also used the scripts that were provided which included the encryption, decryption, English detection, and dictionary text file.

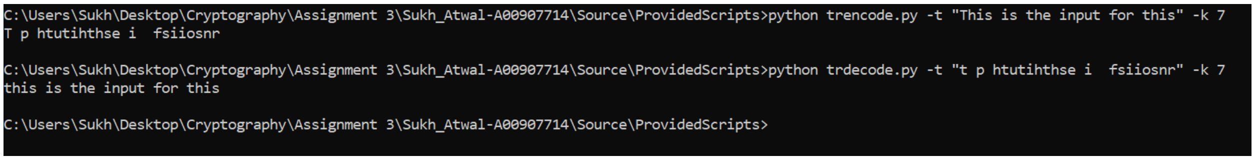
How to Use

If you want to encrypt a message, go to the ‘ProvidedScripts’ in the Source folder and run the trencode.py through the terminal. The arguments are:

-t “Input whatever you want to encrypt here” -k “Enter the key size”

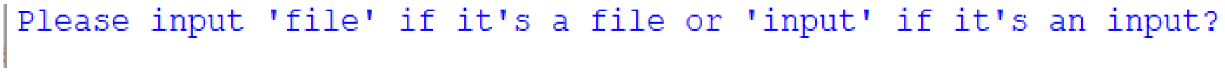
In the same folder, there is also a way to decrypt the encrypted message if you know the key. You would run the trdecode.py through the terminal and use the following arguments:

-t “The encrypted text” -k “The key size”



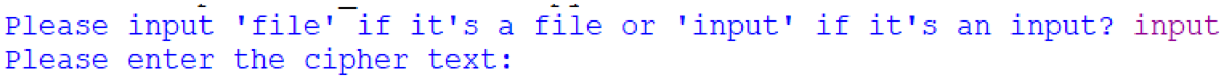
```
C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trencode.py -t "This is the input for this" -k 7
T p htutihthse i  fsiiosnr

C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trdecode.py -t "t p htutihthse i  fsiiosnr" -k 7
this is the input for this

C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>
```

Finally, in the ‘Source’ folder there is a script to break a transposition cipher if you don’t know the key size. You would run the script through the terminal where it would ask if it’s a file or input you want to use.



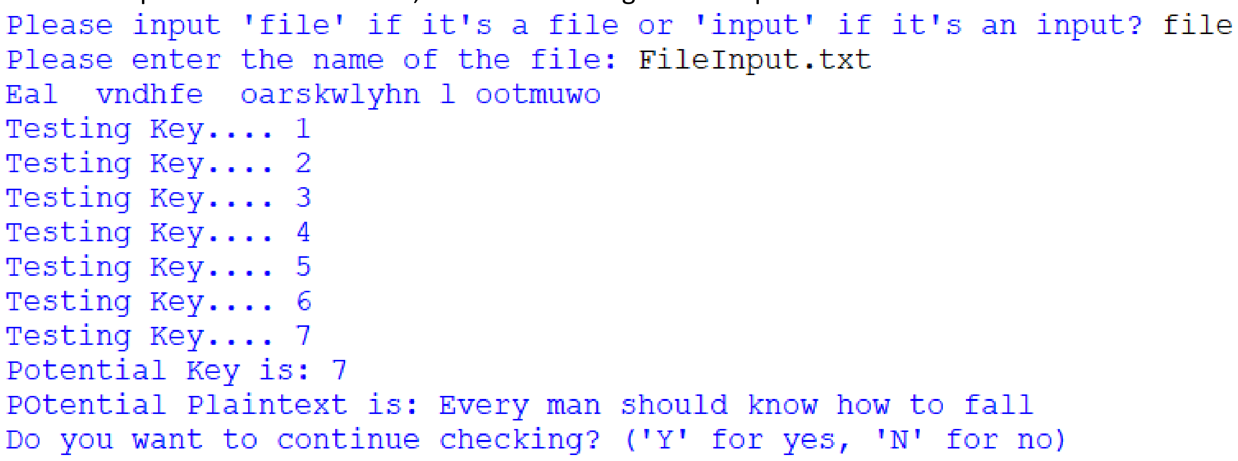
```
Please input 'file' if it's a file or 'input' if it's an input?
```

If we choose input, we would then manually enter or paste the cipher text



```
Please input 'file' if it's a file or 'input' if it's an input? input
Please enter the cipher text:
```

If we choose file, we would enter the file name with the encrypted text. In this case, I already have a file called FileInput.txt which I entered, and the following is the output.

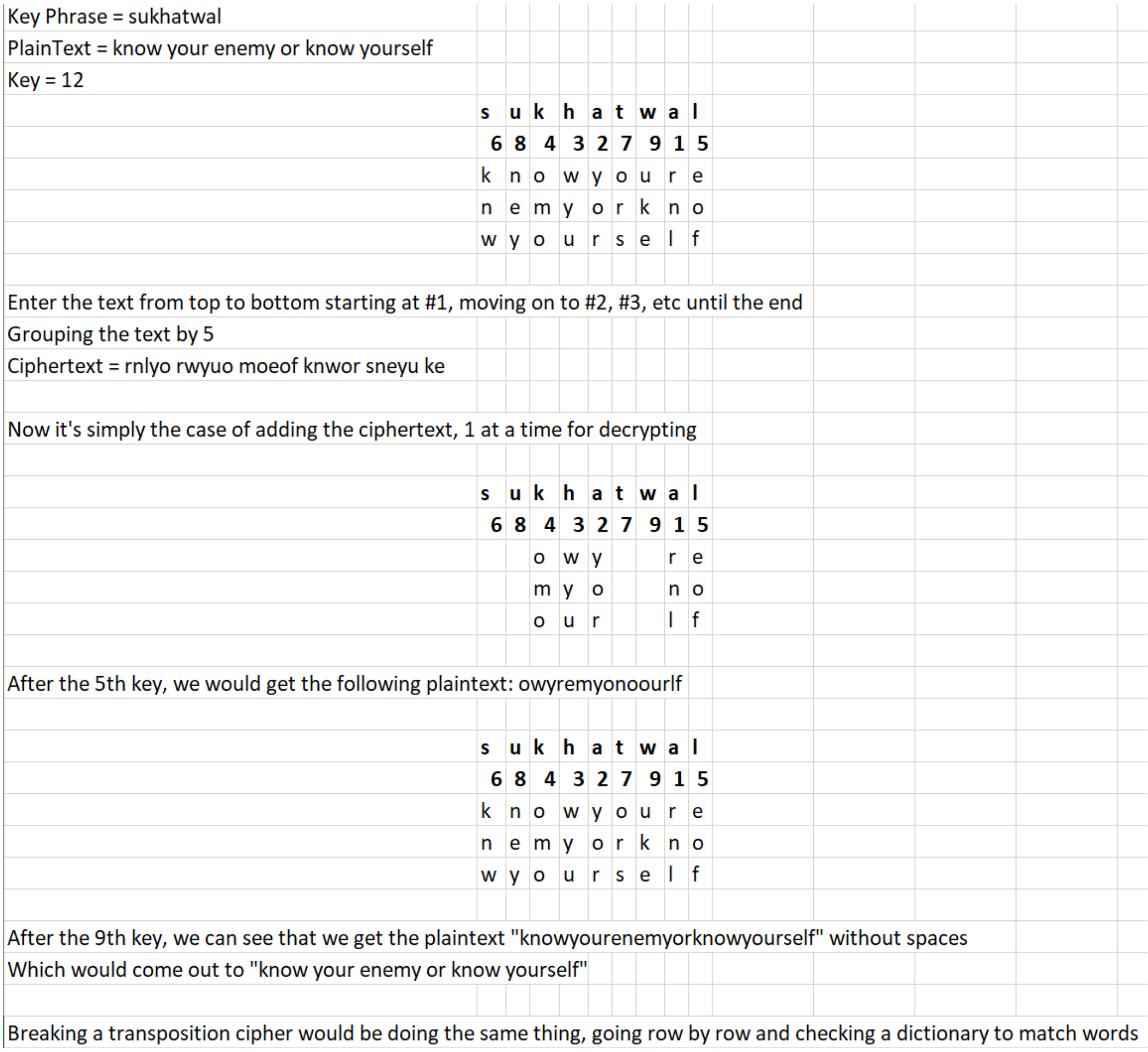


```
Please input 'file' if it's a file or 'input' if it's an input? file
Please enter the name of the file: FileInput.txt
Eal  vndhfe  oarskwlyhn l ootmuwo
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Testing Key.... 7
Potential Key is: 7
POtential Plaintext is: Every man should know how to fall
Do you want to continue checking? ('Y' for yes, 'N' for no)
```

At the end, it’ll ask if you want to continue checking (Y) or stop checking (N).

How it Works - Diagram

The following diagram is an explanation of how the encryption and decryption of transposition ciphers can work.



Key Phrase = sukhatwal

PlainText = know your enemy or know yourself

Key = 12

s	u	k	h	a	t	w	a	l
6	8	4	3	2	7	9	1	5
k	n	o	w	y	o	u	r	e
n	e	m	y	o	r	k	n	o
w	y	o	u	r	s	e	l	f

Enter the text from top to bottom starting at #1, moving on to #2, #3, etc until the end

Grouping the text by 5

Ciphertext = rnlyo rwyuo moeof knwor sneyu ke

Now it's simply the case of adding the ciphertext, 1 at a time for decrypting

s	u	k	h	a	t	w	a	l
6	8	4	3	2	7	9	1	5
		o	w	y			r	e
		m	y	o			n	o
		o	u	r			l	f

After the 5th key, we would get the following plaintext: owyremyonoourlf

s	u	k	h	a	t	w	a	l
6	8	4	3	2	7	9	1	5
k	n	o	w	y	o	u	r	e
n	e	m	y	o	r	k	n	o
w	y	o	u	r	s	e	l	f

After the 9th key, we can see that we get the plaintext "knowyourenemyorknowyourself" without spaces

Which would come out to "know your enemy or know yourself"

Breaking a transposition cipher would be doing the same thing, going row by row and checking a dictionary to match words

Essentially, breaking the transposition cipher would just revolve around an automatic process that tries every key until it finds a combination of letters that matches another combination of letters in a dictionary script that your script would check with.

Testing

My testing setup is a Windows computer running Windows 10 using IDLE as my Python development platform of choice. You can also view all the tests in the 'Videos' folder.

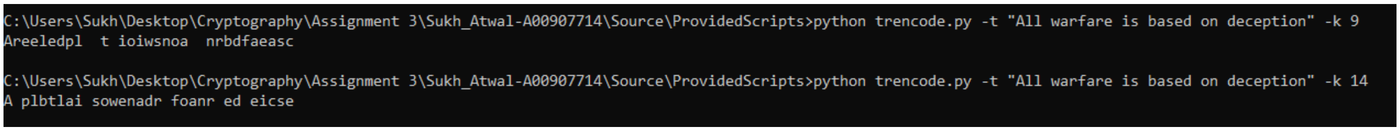
Test #	Tools Used	Expectation	Actuality	Pass or Fail
1	IDLE, CMD	The text should be able to be encrypted and return the same text when decrypted.	The encrypted text returned the same text when decrypted	Pass
2	IDLE	The 'input' option should allow me to input encrypted text and should output the correct decrypted output back.	When we input the cipher, it was successfully decrypted with the same keys.	Pass
3	IDLE	The 'file option should allow me to enter a file with encrypted text in it and should output the correct decrypted output back.	When we had a file with the cipher, it was successfully decrypted with the same keys.	Pass
4	IDLE	The script should not work if the user does not enter 'file' or 'input' and output a message saying to enter the correct inputs	The script asked for the correct inputs when the user input ones that weren't listed	

Test 1

Throughout the rest of the tests, I'll be using the following two texts for breaking the encryption. Here I'll show the encryption and decryption process to ensure the accuracy of it.

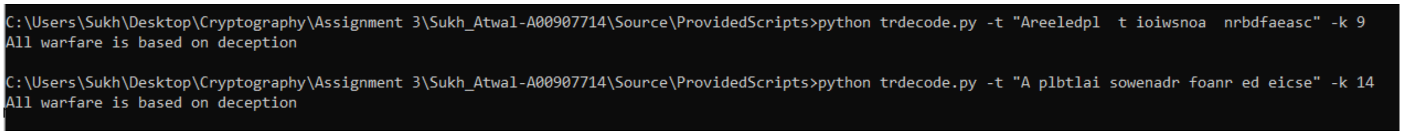
I'll be using two text statements to ensure accuracy. The first will be "All warfare is based on deception" with a key size of 9 and 14.

For the first text statement "All warfare is based on deception", it outputs the following two cipher text:



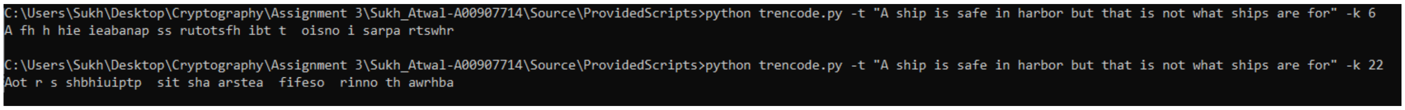
```
C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trencode.py -t "All warfare is based on deception" -k 9
Areeledpl  t ioiwsnoa  nrbdfaeasc

C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trencode.py -t "All warfare is based on deception" -k 14
A plbtlai sowenadr foanr ed eicse
```

Now if I run the decryption script, using the ciphertext corresponding with the key should output the text statement, "All warfare is based on deception".



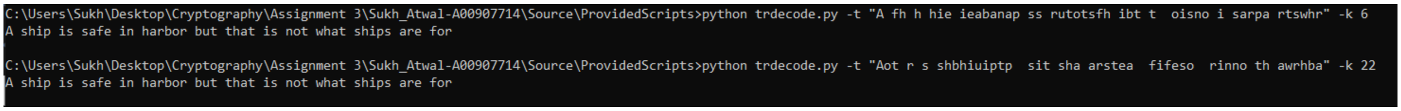
```
C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trdecode.py -t "Areeledpl  t ioiwsnoa  nrbdfaeasc" -k 9
All warfare is based on deception

C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trdecode.py -t "A plbtlai sowenadr foanr ed eicse" -k 14
All warfare is based on deception
```

I'll repeat what I did above but for the second text statement "A ship is safe in harbor but that is not what ships are for" with a key size of 6 and 22.



```
C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trencode.py -t "A ship is safe in harbor but that is not what ships are for" -k 6
A fh h hie ieabanap ss rutotsfh ibt t  oisno i sarpa rtswhr

C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trencode.py -t "A ship is safe in harbor but that is not what ships are for" -k 22
Aot r s shbhiuiptp  sit sha arstea  fifeso  rinno th awrhba
```

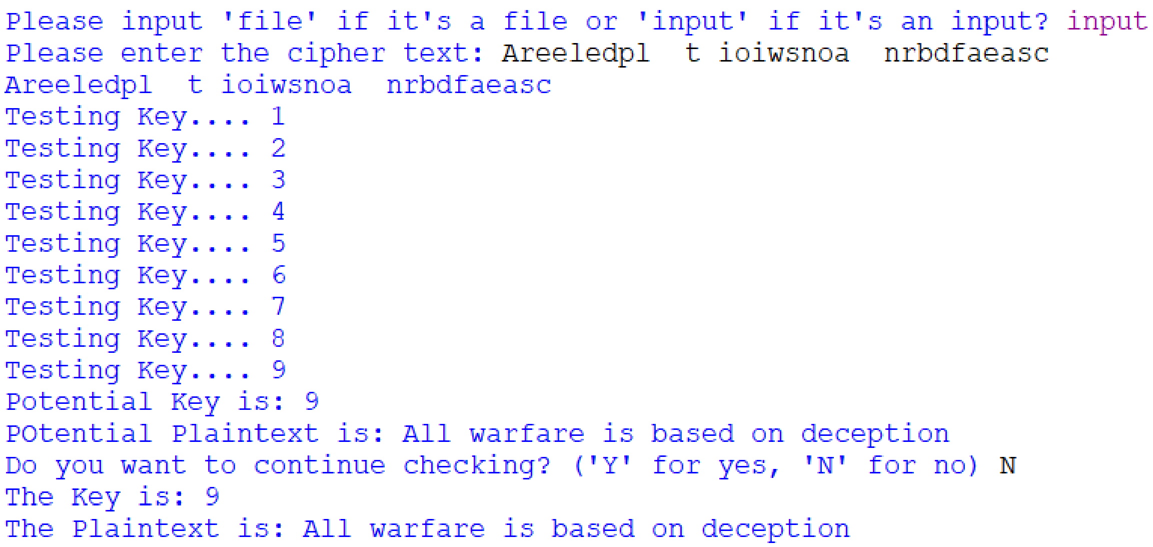
And if I decrypt it, it should output the same text.



```
C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trdecode.py -t "A fh h hie ieabanap ss rutotsfh ibt t  oisno i sarpa rtswhr" -k 6
A ship is safe in harbor but that is not what ships are for

C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\Source\ProvidedScripts>python trdecode.py -t "Aot r s shbhiuiptp  sit sha arstea  fifeso  rinno th awrhba" -k 22
A ship is safe in harbor but that is not what ships are for
```

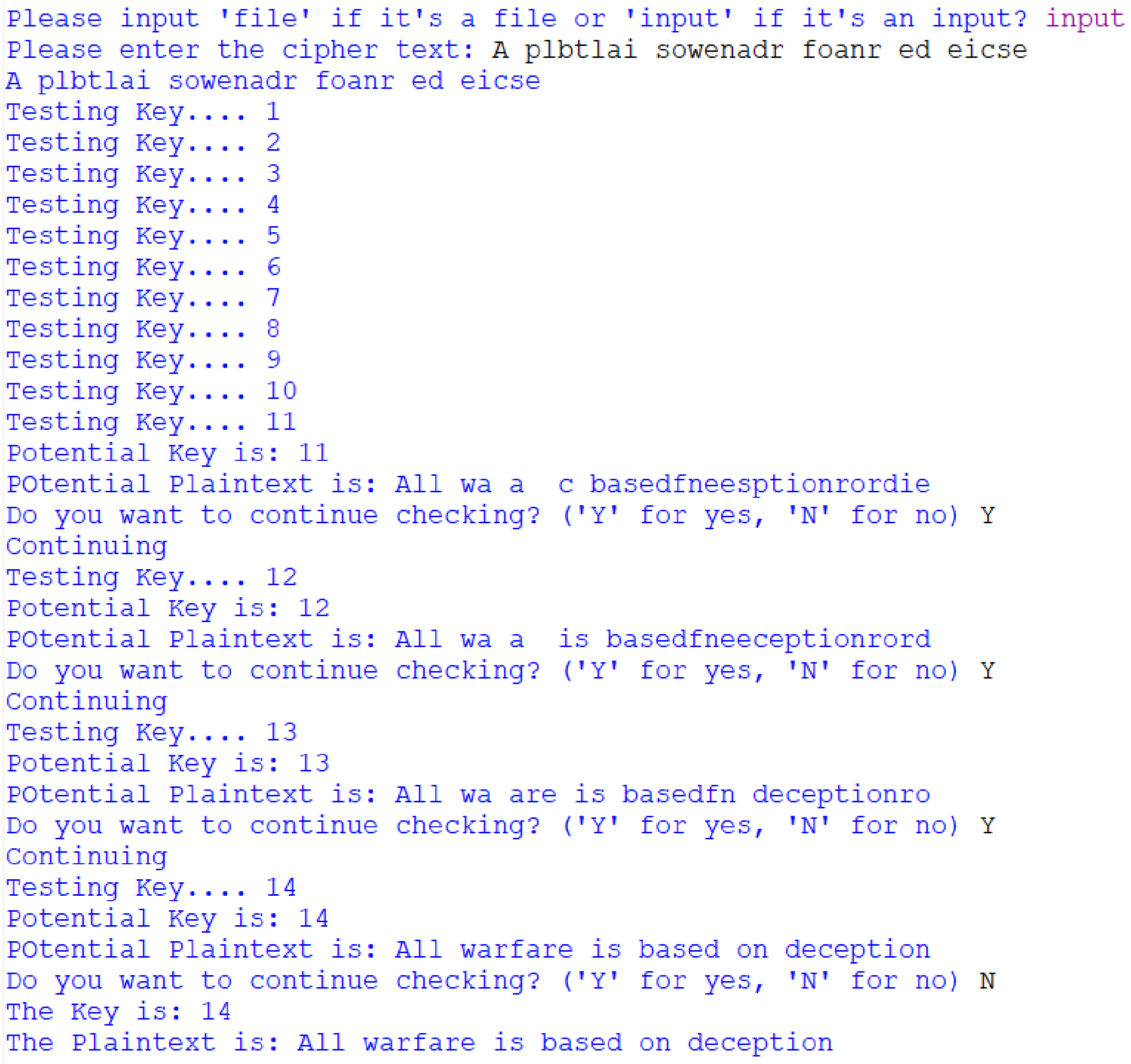
Test 2

I’ll start the script to break the cipher text. Here I’ll be using the ‘input’ option to manually input the 4 encrypted texts.



```
Please input 'file' if it's a file or 'input' if it's an input? input
Please enter the cipher text: Areeledpl  t ioiwsnoa  nrbdfaeasc
Areeledpl  t ioiwsnoa  nrbdfaeasc
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Testing Key.... 7
Testing Key.... 8
Testing Key.... 9
Potential Key is: 9
POtential Plaintext is: All warfare is based on deception
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 9
The Plaintext is: All warfare is based on deception
```

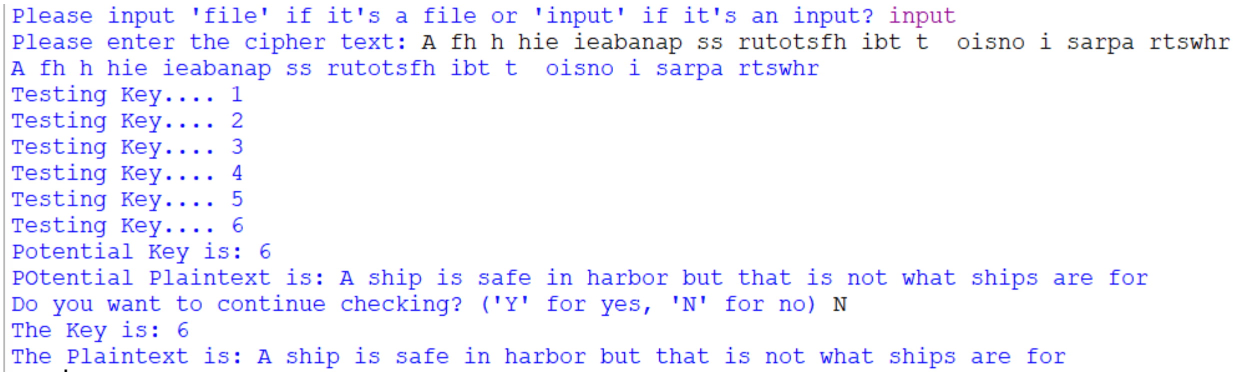
Here is the results of the first cipher text which we created with a key size of 9. Seeing as the plaintext was correct, we didn’t continue checking.



```
Please input 'file' if it's a file or 'input' if it's an input? input
Please enter the cipher text: A plbtlai sowenadr foanr ed eicse
A plbtlai sowenadr foanr ed eicse
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Testing Key.... 7
Testing Key.... 8
Testing Key.... 9
Testing Key.... 10
Testing Key.... 11
Potential Key is: 11
POtential Plaintext is: All wa a  c basedfneesptionrordie
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 12
Potential Key is: 12
POtential Plaintext is: All wa a  is basedfneeceptionrord
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 13
Potential Key is: 13
POtential Plaintext is: All wa are is basedfn deceptionro
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 14
Potential Key is: 14
POtential Plaintext is: All warfare is based on deception
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 14
The Plaintext is: All warfare is based on deception
```

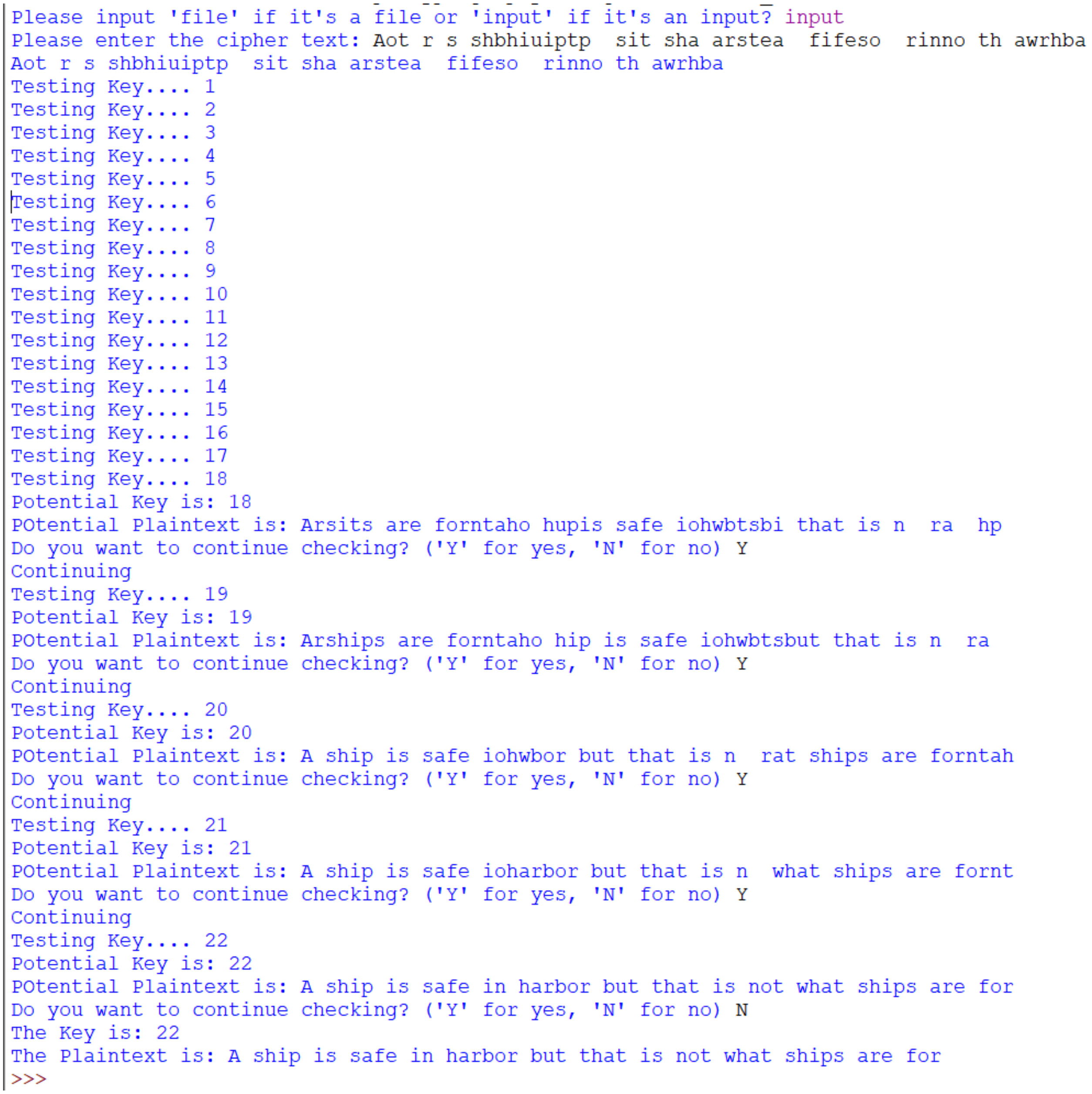
Here, we can see that the plaintext was random letters and wasn't outputting any sensible words, so we continued checking. Eventually, we got the correct output with a key of 14 which matches.

Now we'll move to the second statement.



```
Please input 'file' if it's a file or 'input' if it's an input? input
Please enter the cipher text: A fh h hie ieabanap ss rutotsfh ibt t  oisno i sarpa rtswhr
A fh h hie ieabanap ss rutotsfh ibt t  oisno i sarpa rtswhr
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Potential Key is: 6
POtential Plaintext is: A ship is safe in harbor but that is not what ships are for
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 6
The Plaintext is: A ship is safe in harbor but that is not what ships are for
```

The plaintext is correct, and it got it with a key of 6 which matches what we did when creating it.

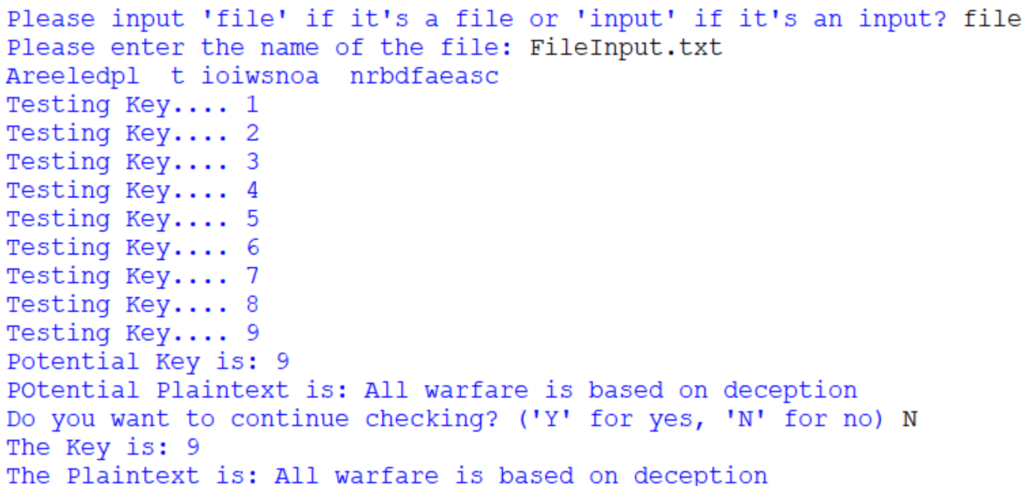


```
Please input 'file' if it's a file or 'input' if it's an input? input
Please enter the cipher text: Aot r s shbhiuiptp  sit sha arstea  fifeso  rinno th awrhba
Aot r s shbhiuiptp  sit sha arstea  fifeso  rinno th awrhba
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Testing Key.... 7
Testing Key.... 8
Testing Key.... 9
Testing Key.... 10
Testing Key.... 11
Testing Key.... 12
Testing Key.... 13
Testing Key.... 14
Testing Key.... 15
Testing Key.... 16
Testing Key.... 17
Testing Key.... 18
Potential Key is: 18
POtential Plaintext is: Arsits are forntaho hupis safe iohwbtsbi that is n  ra  hp
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 19
Potential Key is: 19
POtential Plaintext is: Arships are forntaho hip is safe iohwbtsbut that is n  ra
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 20
Potential Key is: 20
POtential Plaintext is: A ship is safe iohwbor but that is n  rat ships are forntah
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 21
Potential Key is: 21
POtential Plaintext is: A ship is safe ioharbor but that is n  what ships are fornt
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 22
Potential Key is: 22
POtential Plaintext is: A ship is safe in harbor but that is not what ships are for
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 22
The Plaintext is: A ship is safe in harbor but that is not what ships are for
>>>
```

Here, we see a similar thing that happened before where the plaintext doesn’t initially match what we put in however it’s also still random letters and doesn’t create any legible sentences until key 22.

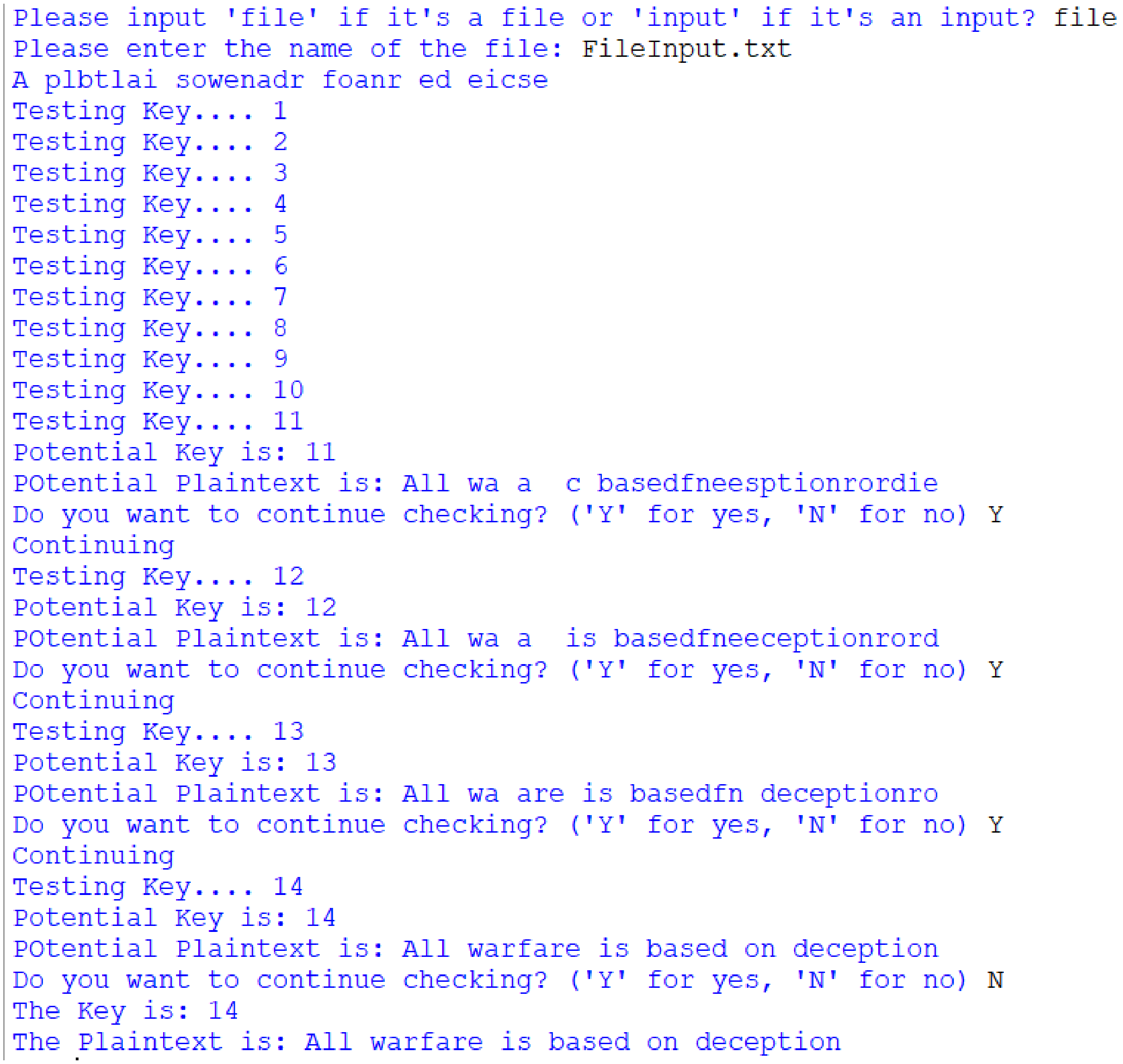
Test 3

Here I’ll be doing the same idea as above however I’ll be using the ‘file’ option which allows you to break a transposition cipher that’s in the file. I’ll be using a file called FileInput.txt in this scenario.



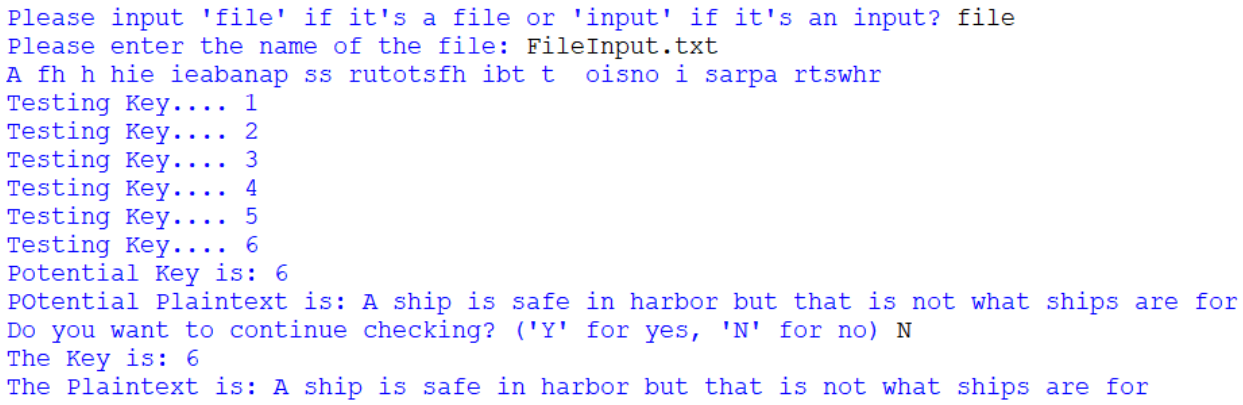
```
Please input 'file' if it's a file or 'input' if it's an input? file
Please enter the name of the file: FileInput.txt
Areeledpl  t ioiwsnoa  nrbdfaeasc
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Testing Key.... 7
Testing Key.... 8
Testing Key.... 9
Potential Key is: 9
POtential Plaintext is: All warfare is based on deception
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 9
The Plaintext is: All warfare is based on deception
```

We can see that it matches the rest of the tests, where it correctly gets the plaintext after 9 attempts.



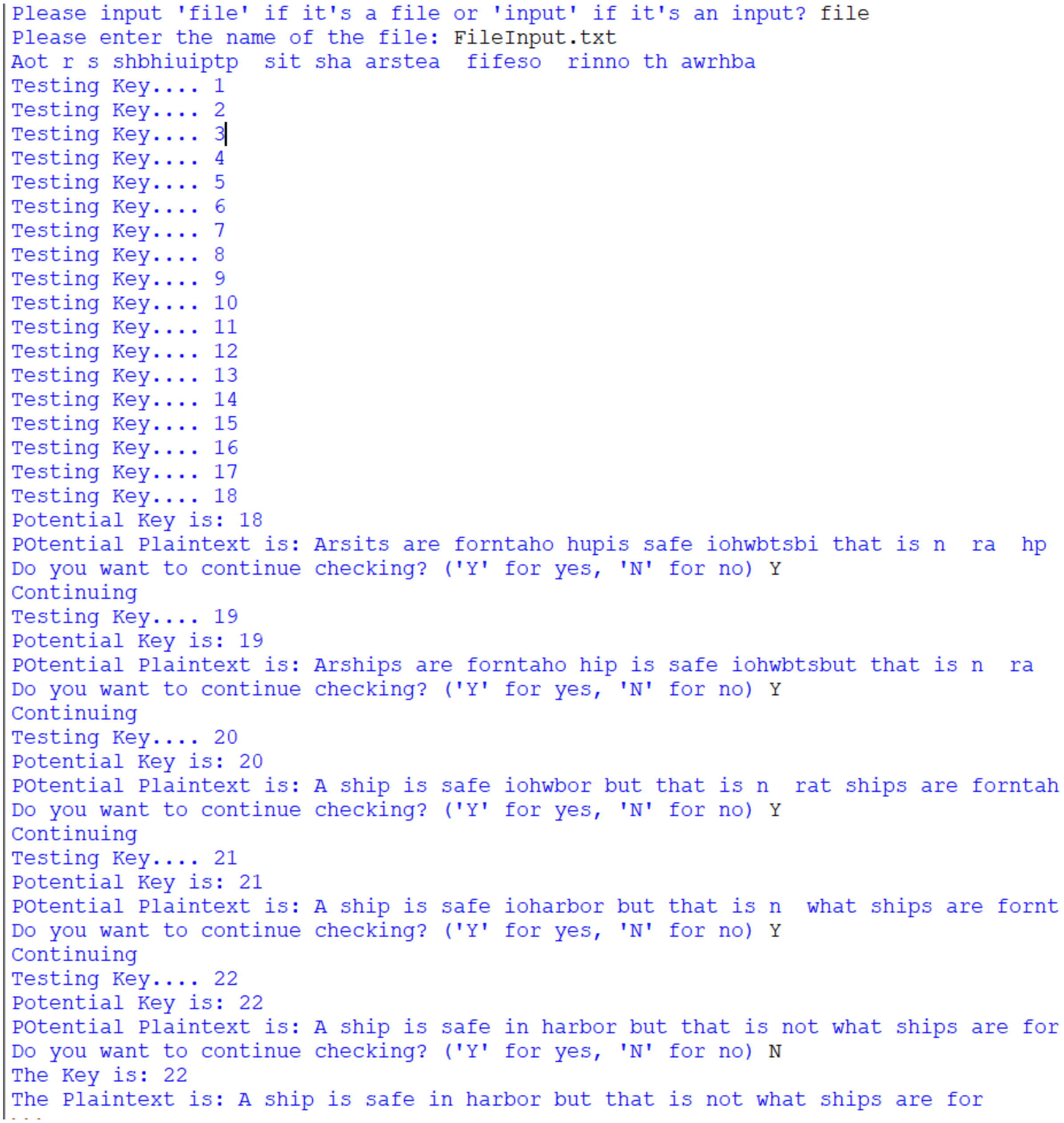
```
Please input 'file' if it's a file or 'input' if it's an input? file
Please enter the name of the file: FileInput.txt
A plbtlai sowenadr foanr ed eicse
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Testing Key.... 7
Testing Key.... 8
Testing Key.... 9
Testing Key.... 10
Testing Key.... 11
Potential Key is: 11
POtential Plaintext is: All wa a  c basedfneesptionrordie
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 12
Potential Key is: 12
POtential Plaintext is: All wa a  is basedfneeceptionrord
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 13
Potential Key is: 13
POtential Plaintext is: All wa are is basedfn deceptionro
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 14
Potential Key is: 14
POtential Plaintext is: All warfare is based on deception
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 14
The Plaintext is: All warfare is based on deception
```

Similarly, we can see it some of the initial plaintexts are just random letters and words strung together up until Key 14, where it outputs the correct plaintext.



```
Please input 'file' if it's a file or 'input' if it's an input? file
Please enter the name of the file: FileInput.txt
A fh h hie ieabanap ss rutotsfh ibt t  oisno i sarpa rtswhr
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Potential Key is: 6
POtential Plaintext is: A ship is safe in harbor but that is not what ships are for
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 6
The Plaintext is: A ship is safe in harbor but that is not what ships are for
```

For the second text statement we encrypted, the results are as expected with the correct plaintext for Key 6.

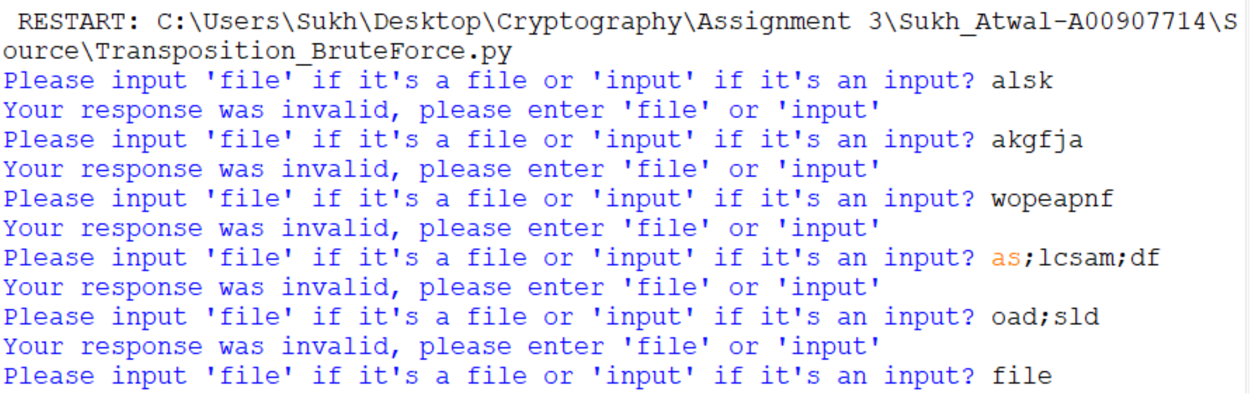


```
Please input 'file' if it's a file or 'input' if it's an input? file
Please enter the name of the file: FileInput.txt
Aot r s shbhiuiptp  sit sha arstea  fifeso  rinno th awrhba
Testing Key.... 1
Testing Key.... 2
Testing Key.... 3
Testing Key.... 4
Testing Key.... 5
Testing Key.... 6
Testing Key.... 7
Testing Key.... 8
Testing Key.... 9
Testing Key.... 10
Testing Key.... 11
Testing Key.... 12
Testing Key.... 13
Testing Key.... 14
Testing Key.... 15
Testing Key.... 16
Testing Key.... 17
Testing Key.... 18
Potential Key is: 18
POtential Plaintext is: Arsits are forntaho hupis safe iohwbtsbi that is n  ra  hp
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 19
Potential Key is: 19
POtential Plaintext is: Arships are forntaho hip is safe iohwbtsbut that is n  ra
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 20
Potential Key is: 20
POtential Plaintext is: A ship is safe iohwbor but that is n  rat ships are forntah
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 21
Potential Key is: 21
POtential Plaintext is: A ship is safe ioharbor but that is n  what ships are fornt
Do you want to continue checking? ('Y' for yes, 'N' for no) Y
Continuing
Testing Key.... 22
Potential Key is: 22
POtential Plaintext is: A ship is safe in harbor but that is not what ships are for
Do you want to continue checking? ('Y' for yes, 'N' for no) N
The Key is: 22
The Plaintext is: A ship is safe in harbor but that is not what ships are for
```

Once again, random letters and words initially for the plaintext but eventually gives the correct plaintext with a Key of 22.

Test 4

The script should repeatedly ask the user to correctly enter the commands, either ‘input’ if it’s a user input or ‘file’ if it’s a file.



```
 RESTART: C:\Users\Sukh\Desktop\Cryptography\Assignment 3\Sukh_Atwal-A00907714\S
ource\Transposition_BruteForce.py
Please input 'file' if it's a file or 'input' if it's an input? alsk
Your response was invalid, please enter 'file' or 'input'
Please input 'file' if it's a file or 'input' if it's an input? akgfja
Your response was invalid, please enter 'file' or 'input'
Please input 'file' if it's a file or 'input' if it's an input? wopeapnf
Your response was invalid, please enter 'file' or 'input'
Please input 'file' if it's a file or 'input' if it's an input? as;lcsam;df
Your response was invalid, please enter 'file' or 'input'
Please input 'file' if it's a file or 'input' if it's an input? oad;sld
Your response was invalid, please enter 'file' or 'input'
Please input 'file' if it's a file or 'input' if it's an input? file
```

As you can see, if the user enters random gibberish, the script asks if the user can input the correct commands. This continues until the user enters them.