Assignment 3

● Graded

Student

Mohammed Jawed

Total Points

100 / 100 pts

Question 1

Commands

10 / 10 pts

✓ **+ 10 pts** Correct

+ 0 pts Inorrect

Question 2

Cryptosystem

5 / 5 pts

✓ **+ 5 pts** Correct

+ 0 pts Incorrect

Question 3

Analysis

80 / 80 pts

✓ **+ 10 pts** Conducting Frequency analysis

✓ **+ 10 pts** Concluding about permutation in the cipher with proper justification

✓ **+ 10 pts** Explaining and Concluding about the block length 5 .

✓ **+ 10 pts** Mentioning the correct permutation sequence

✓ **+ 10 pts** Mentioning the details of finding the correct permutation sequence of 1st and 2nd position

✓ **+ 10 pts** Mentioning the details of finding the correct permutation sequence of the remaining position in the block.

✓ **+ 10 pts** Step by Step explanation of the whole deciphering using the blocklength alphabets/ words which are meaningful

✓ **+ 10 pts** Taking care of the spaces/punctuation

+ 0 pts incorrect/no submission

Question 4

Password

5 / 5 pts

+ 0 pts Incorrect

✓ **+ 5 pts** Correct

Question 5

Code

0 / 0 pts

✓ **+ 0 pts** Correct

Q1 Commands

10 Points

List the commands used in the game to reach the first ciphertext.

-enter -enter -pick -back -put -back
-give -back -back -thrnxxtzy -read

Q2 Cryptosystem

5 Points

What cryptosystem was used in this level?

Substitution-Permutation network

Q3 Analysis

80 Points

What tools and observations were used to figure out the cryptosystem and the password? (Explain in less than 1000 lines)

Note: I have segregated the observation in various section based on various parameters.

Attachments: 1. Complete commands log recorded during game
2. code used in this Assigment.
3. Screen shot of letters weightage and their frequency

The happy path flow with command has been recorded during level 3 Assigment are attached with this analysis (see the attachment part).

The cyphertext encountered during Assigment is:

```
qmnjvsa nv wewc flct vprj tj tvvplvl fv xja vqildhc
xmlnvc nacyclpa fc gyt vfvw. fv wgqyp, pqq pqcs y wsq
rx qmnjvafy cgv tlvhf cw tyl aeuq fv xja tkbv cqnsqs.
lhf avawnc cv eas fuqb qvq tc yllrqr xxwa cfy. psdc uqf
avrqc gefq pyat trac xwv taa wwd dv eas flcbq. vd trawm
vupq quw x decgqcwt, yq yafl vlqs yqklhq! snafq vml
lhvqpawr nqg_vfusr_ec_wawy qp fn wgawdgf.
```

```
> the_magic_of_wand
```

- Cryptographic Analysis Process
  - -- Initial Examination and Frequency Analysis

I began with a raw ciphertext consisting of 356 characters. After cleansing it of punctuation and special characters, I was left with 287 characters. Further exclusion of a segment identified as a probable password (as we have experienced this during Assigment-1 and Assigment-2)[nqg_vfusr_ec_wawy] reduced this to 270 characters.

A frequency analysis (code attached) was conducted on these characters. The character distribution closely resembled that of normal English text, suggesting that a simple substitution cipher might have been employed. The even distribution also supported the hypothesis of a sophisticated encoding technique, suggesting a layered encryption approach—likely a mix of substitution and permutation.

- Establishing Cipher Characteristics

1. Block Length Determination:
Considering the character count (270), I analyzed potential block sizes by their factorization of 270, considering permutation ciphers used in cryptographic puzzles. The factors include 1, 2, 3, 5, 6, 9, 10, 15, 18, 27, 30, 45, 54, 90, 135, and 270. A block length of 5 was chosen based on its cryptographic suitability and the absence of padding needs. The choice of a 5-character block was substantiated by the need to align with the known password length from prior puzzles and the divisibility of 270 by 5, which allows for uniform block processing without the need for padding.

2. Permutation Analysis:
My decryption strategy was based on the hypothesis that the cipher involved block permutations. By comparing segments possibly containing instructions or passwords with known plaintext structures from previous puzzles, By examining phrases presumed to be part of the ciphertext, such as potential passwords or instructions, a pattern in the arrangement of letters within blocks began to emerge.

```
jREAKER OF THIS CODE WItt jE jtESSED jx THE SidEAKx
SPIRIT RESIDIyG Iy THE HOtE. GO AHEAD, AyD FIyD A WAx
OF jREAKIyG THE SPEtt Oy HIu CAST jx THE EbIt kAFFAR.
THE SPIRIT OF THE CAbE uAy IS AtWAxS WITH xOd. FIyD THE
uAGIC WAyD THAT WItt tET xOd OdT OF THE CAbES. IT WOdtD
uAKE xOd A uAGICIAy, yO tESS THAy kAFFAR! SPEAK THE
PASSWORD THE_uAGIC_OF_WAyD TO GO THR
```

- Decrypting the Cipher

1. First and Second Position Permutation:
My decryption efforts focused initially on known plaintext structures within the ciphertext, such as "speak the password".
Observations of these segments revealed that the expected end characters were instead at the beginning, leading me to adjust the positions accordingly.

2.Remaining Positions and Complete Sequence:
With the initial positions identified, the decryption of the remaining positions was approached similarly. Aligning the observed output with expected plaintext confirmed the full permutation sequence, which was established as (3, 2, 4, 0, 1).

3. Comprehensive Block Decryption:
With the permutation sequence clarified, the entire text was segmented into

5-character blocks. Each block underwent permutation adjustment followed by a character substitution based on my earlier frequency analysis and identified patterns. This methodical decryption clarified the text and exposed the hidden message.

4. Integration of Spaces and Punctuation:
Given the removal of spaces and punctuation for the decryption process, these were reintegrated post-decryption to restore the text's readability and grammatical integrity. Logical breaks and sentence structures from the decrypted text guided this reintegration.

"breaker of this code will be blessed by the squeaky spirit residing in the hole.
go ahead, and find away of breaking the spell on him cast by the evil jaffar. the
spirit of the cave man is always with you. find the magic wand that will let you
out of the caves. it would make you a magician, no less than jaffar! to go through, speak the password the_magic_of_wand."

At the end the decryption process successfully rendered a meaningful and coherent plaintext, confirming the effectiveness of the identified decryption strategy. The text included specific instructions and a password, "the_magic_of_wand", crucial for advancing in the cryptographic challenge.

The comprehensive analysis of assigement-3 not only showcased the complexity of the cipher but also demonstrated a systematic approach to deciphering multi-technique encrypted messages, underscoring the need for a nuanced understanding of both permutation and substitution ciphers in cryptanalysis.

Q4 Password

5 Points

What was the final command used to clear this level?

the_magic_of_wand

Q5 Code

0 Points

Upload any code that you have used to solve this level.

putty-Game3.log

Download

```
=~=~=~=~=~=~=~=~=~=~=~= PuTTY log 2024.05.10 12:55:26 =~=~=~=~=~=~=~=~=~=~=~=
3




The chamber is completely dark. You quickly pull out the
matchbox and light a stick ...

The light fills up the chamber slowly. By now you are used
to dim lights and so see things immediately. The chamber is,
like the previous ones, made by carving through the rocks.
Its floor is somewhat uneven, but there are no boulders here.
There seems to be a constant rumbling sound in the background.
You could see some odd shapes lying on the floor in a corner.
Becoming curious, you move towards them and all of a sudden,
freeze in your tracks. These are human skeletons!! One of
them has both its hands (whatever is left of it) pointing
upwards as if pleading something. The thought strikes your
mind that perhaps these people could not get past the chamber
and just died! Clearing your mind of negative thoughts, you
hastily withdraw and start looking around. you realize that there
is another door in one side of the chamber (the same side where
the skeletons lie, in your panic you did not notice this earlier).
And the door is not closed! Perhaps there is another chamber here.
You decide to investigate ...

> enter




This is a small chamber. The rumbling sound become louder
here. The floor is full of small rocks. There is a stale,
and somewhat bad, odour here. There is a small door to your
left from which you entered. You throw the partially burnt
matchstick down and light another one.

You notice a large hole in the ground. Next to it, there is a very
small hole in the ground, barely enough to put your hands in it.
Going closer, you realize that the large hole opens to a small,
dark underground chamber. Then you catch a glimpse of something
shiny inside the small hole.

> enter



```

```

The rumbling sound is very loud here. It is very stickly
and smelly too. You want to quickly get out of this place.
Steeling yourself, you begin to investigate. It is a very
small opening. The floor is muddy. You see some mushrooms
growing out of the floor.

> pick





You pluck mushrooms from the floor. They are smelly!


Press c to continue> c


Press c to continue> c




The rumbling sound is very loud here. It is very stickly
and smelly too. You want to quickly get out of this place.
Steeling yourself, you begin to investigate. It is a very
small opening. The floor is muddy. You see some mushrooms
growing out of the floor.

> back




This is a small chamber. The rumbling sound become louder
here. The floor is full of small rocks. There is a stale,
and somewhat bad, odour here. There is a small door to your
left from which you entered. You throw the partially burnt
matchstick down and light another one.

You notice a large hole in the ground. Next to it, there is a very
small hole in the ground, barely enough to put your hands in it.
Going closer, you realize that the large hole opens to a small,
dark underground chamber. Then you catch a glimpse of something
shiny inside the small hole.

> put




```

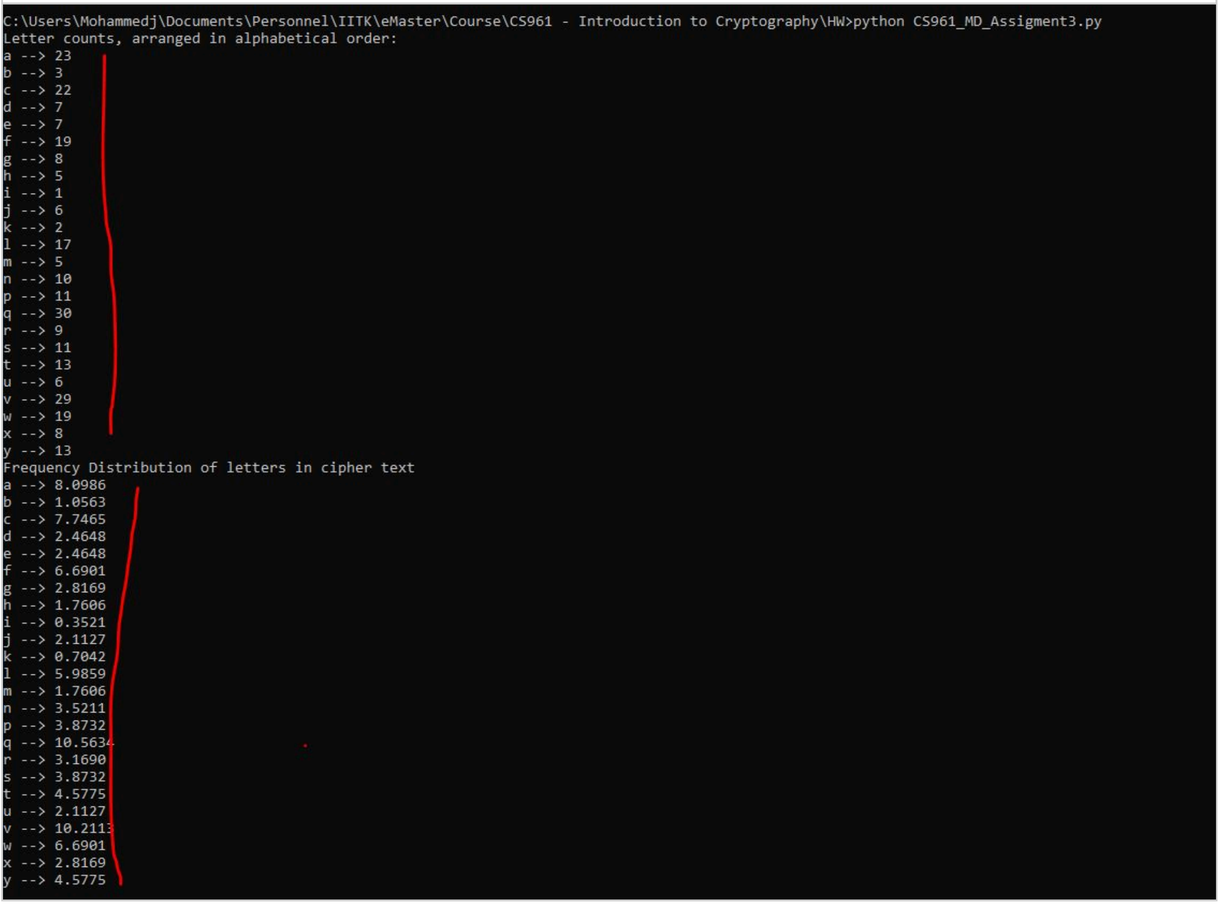
```
You cry out in pain! Someone has bit your hand!!

> back




This is a small chamber. The rumbling sound become louder
here. The floor is full of small rocks. There is a stale,
and somewhat bad, odour here. There is a small door to your
left from which you entered. You throw the partially burnt
matchstick down and light another one.

You notice a large hole in the ground. Next to it, there is a very
small hole in the ground, barely enough to put your hands in it.
Going closer, you realize that the large hole opens to a small,
dark underground chamber. Then you catch a glimpse of something
shiny inside the small hole.

> give




You take some mushrooms in your hand and put it in the hole.
Someone grabs the mushrooms from your hand! You then hear
chomping sound as if they are being quickly eaten. After
a while, the sounds cease ...

You figure that perhaps some rat is sitting inside the hole
eating mushrooms. Suddenly, you hear a squeaky voice
speaking from inside the hole!

"Oh, thank you very much for the mushrooms! I have been hungry
for so long!! I am a poor spirit trapped inside this hole by an
evil man. Maybe you can help me be free ... (sigh) oh, forget
it. I'll help you pass this chamber though. Speak out the magic
words ``thrnxxtzy'' for the hidden door to become visible. The door
lies hidden in the main chamber."

> back




This is a small chamber. The rumbling sound become louder
here. The floor is full of small rocks. There is a stale,
and somewhat bad, odour here. There is a small door to your
left from which you entered. You throw the partially burnt
matchstick down and light another one.

You notice a large hole in the ground. Next to it, there is a very
```

```
small hole in the ground, barely enough to put your hands in it.
Going closer, you realize that the large hole opens to a small,
dark underground chamber. Then you catch a glimpse of something
shiny inside the small hole.

> back




The chamber is completely dark. You quickly pull out the
matchbox and light a stick ...

The light fills up the chamber slowly. By now you are used
to dim lights and so see things immediately. The chamber is,
like the previous ones, made by carving through the rocks.
Its floor is somewhat uneven, but there are no boulders here.
There seems to be a constant rumbling sound in the background.
You could see some odd shapes lying on the floor in a corner.
Becoming curious, you move towards them and all of a sudden,
freeze in your tracks. These are human skeletons!! One of
them has both its hands (whatever is left of it) pointing
upwards as if pleading something. The thought strikes your
mind that perhaps these people could not get past the chamber
and just died! Clearing your mind of negative thoughts, you
hastily withdraw and start looking around. you realize that there
is another door in one side of the chamber (the same side where
the skeletons lie, in your panic you did not notice this earlier).
And the door is not closed! Perhaps there is another chamber here.
You decide to investigate ...

> thrnxxtzy




A door appears in front the front wall! So does a glass panel next to it!!

> read




qmnjvsa nv wewc flct vprj tj tvvplvl fv xja vqildhc
xmlnvc nacyclpa fc gyt vfvw. fv wgqyp, pqq pqcs y wsq
rx qmnjvafy cgv tlvhf cw tyl aeuq fv xja tkbv cqnsqs.
lhf avawnc cv eas fuqb qvq tc yllrqr xxwa cfy. psdc uqf
avrqc gefq pyat trac xwv taa wwd dv eas flcbq. vd trawm
vupq quw x decgqcwt, yq yafl vlqs yqklhq! snafq vml
lhvqpawr nqg_vfusr_ec_wawy qp fn wgawdgf.

> the_magic_wand
```

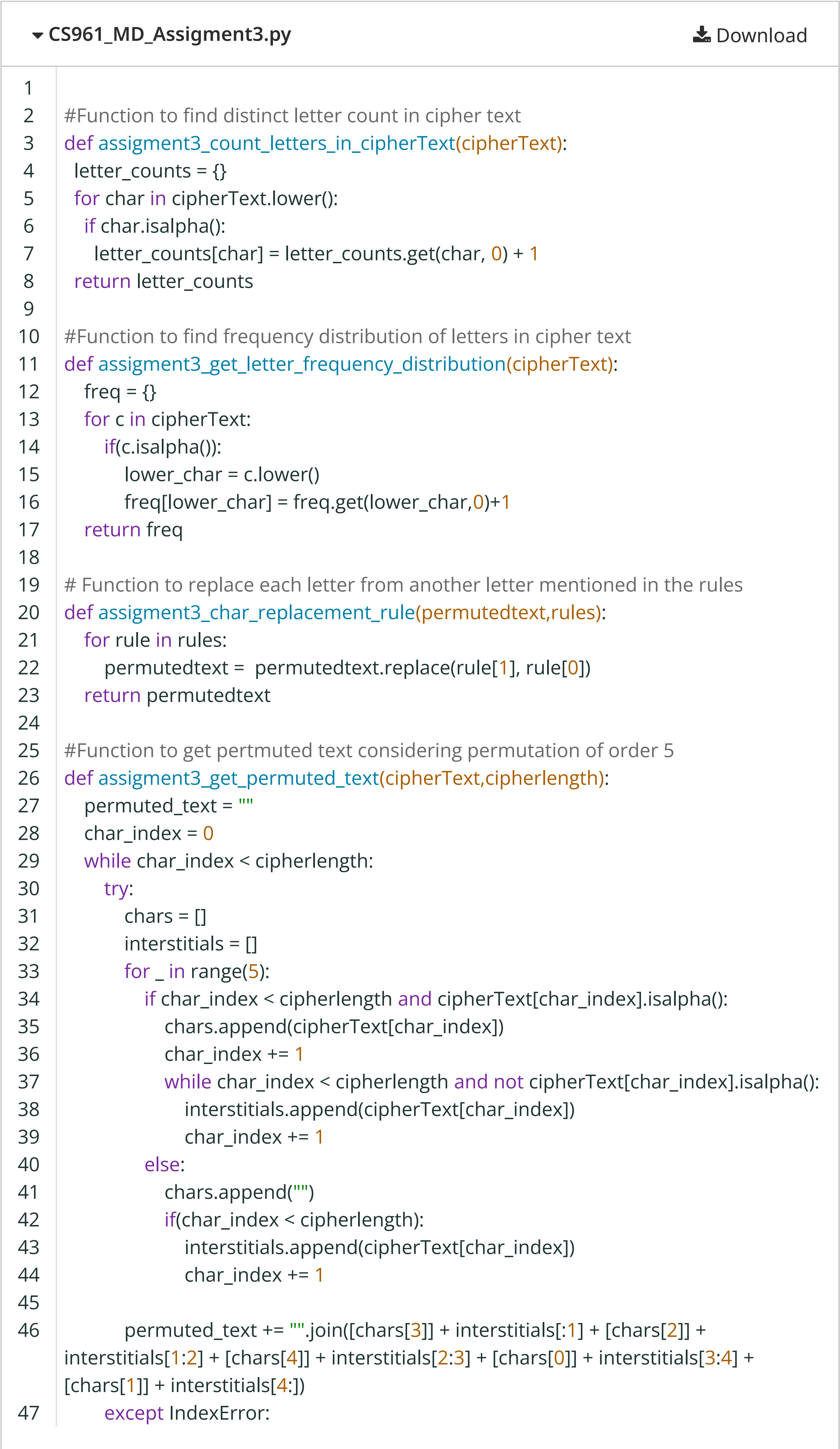
```





Unknown command the_magic_wand!


Press c to continue> c




qmnjvsa nv wewc flct vprj tj tvvplvl fv xja vqildhc
xmlnvc nacyclpa fc gyt vfvw. fv wgqyp, pqq pqcs y wsq
rx qmnjvafy cgv tlvhf cw tyl aeuq fv xja tkbv cqnsqs.
lhf avawnc cv eas fuqb qvq tc yllrqr xxwa cfy. psdc uqf
avrqc gefq pyat trac xwv taa wwd dv eas flcbq. vd trawm
vupq quw x decgqcwt, yq yafl vlqs yqklhq! snafq vml
lhvqpawr nqg_vfusr_ec_wawy qp fn wgawdgf.

> the_magic_of_wand



You enter a narrow passage with very dim light.
The passage is not very long, you can see the other
exit in front of you.
You notice names written all around:
admin suryakant23 mjawed23 raghavd23 priyankarb23




Press c to continue> c




Before you make a move, there is a deafning sound of something crashing
down followed by a lot of dust. As the dust clears, you see that the
passage has been blocked on all sides by rocks fallen from the roof.
You try to find a way around without any success. You notice something
written on one of the fallen rocks. Lighting a matchstick, you
read:

Take a break. It is too early to go to the next level.
> exit

```

▾ Frequency_Order.JPG

Download



```
C:\Users\Mohammedj\Documents\Personnel\IITK\eMaster\Course\CS961 - Introduction to Cryptography\HW>python CS961_MD_Assigment3.py
Letter counts, arranged in alphabetical order:
a --> 23
b --> 3
c --> 22
d --> 7
e --> 7
f --> 19
g --> 8
h --> 5
i --> 1
j --> 6
k --> 2
l --> 17
m --> 5
n --> 10
p --> 11
q --> 30
r --> 9
s --> 11
t --> 13
u --> 6
v --> 29
w --> 19
x --> 8
y --> 13
Frequency Distribution of letters in cipher text
a --> 8.0986
b --> 1.0563
c --> 7.7465
d --> 2.4648
e --> 2.4648
f --> 6.6901
g --> 2.8169
h --> 1.7606
i --> 0.3521
j --> 2.1127
k --> 0.7042
l --> 5.9859
m --> 1.7606
n --> 3.5211
p --> 3.8732
q --> 10.5634
r --> 3.1690
s --> 3.8732
t --> 4.5775
u --> 2.1127
v --> 10.2113
w --> 6.6901
x --> 2.8169
y --> 4.5775
```



```python

#Function to find distinct letter count in cipher text
def assigment3_count_letters_in_cipherText(cipherText):
  letter_counts = {}
  for char in cipherText.lower():
    if char.isalpha():
      letter_counts[char] = letter_counts.get(char, 0) + 1
  return letter_counts

#Function to find frequency distribution of letters in cipher text
def assigment3_get_letter_frequency_distribution(cipherText):
    freq = {}
    for c in cipherText:
        if(c.isalpha()):
            lower_char = c.lower()
            freq[lower_char] = freq.get(lower_char,0)+1
    return freq

# Function to replace each letter from another letter mentioned in the rules
def assigment3_char_replacement_rule(permutedtext,rules):
    for rule in rules:
        permutedtext =  permutedtext.replace(rule[1], rule[0])
    return permutedtext

#Function to get pertmuted text considering permutation of order 5
def assigment3_get_permuted_text(cipherText,cipherlength):
    permuted_text = ""
    char_index = 0
    while char_index < cipherlength:
        try:
            chars = []
            interstitials = []
            for _ in range(5):
                if char_index < cipherlength and cipherText[char_index].isalpha():
                    chars.append(cipherText[char_index])
                    char_index += 1
                    while char_index < cipherlength and not cipherText[char_index].isalpha():
                        interstitials.append(cipherText[char_index])
                        char_index += 1
                else:
                    chars.append("")
                    if(char_index < cipherlength):
                        interstitials.append(cipherText[char_index])
                        char_index += 1

            permuted_text += "".join([chars[3]] + interstitials[:1] + [chars[2]] + interstitials[1:2] + [chars[4]] + interstitials[2:3] + [chars[0]] + interstitials[3:4] + [chars[1]] + interstitials[4:])
        except IndexError:
            break
    return permuted_text

#Function to reorder the letters in decrypted text.

def assigment3_re_order_letters(cipherText,permutedtext):
    pos = 0
    decryptedText = ""
    length =len(cipherText.split(' '));
    for l in range(length):
        s = cipherText.split(' ')[l]

        for i in range(len(s)):
            sl = len(s)
            if(permutedtext[pos].endswith("\n")): break
            if(permutedtext[pos] == ' '):
                pos+=1
            decryptedText += permutedtext[pos]
            pos += 1
        decryptedText += ' '
    return   decryptedText

# main function
def main():
    cipherText = "qmnjvsa nv wewc flct vprj tj tvvplvl fv xja vqildhc xmlnvc nacyclpa fc gyt vfvw. fv wgqyp, pqq pqcs y wsq rx qmnjvafy cgv tlvhf cw tyl aeuq fv xja tkbv cqnsqs. lhf avawnc cv eas fuqb qvq tc yllrqr xxwa cfy. psdc uqf avrqc gefq pyat trac xwv taa wwd dv eas flcbq. vd trawm vupq quw x decgqcwt, yq yafl vlqs yqklhq! snafq vml lhvqpawr nqg_vfusr_ec_wawy qp fn wgawdgf."
    cipherlen = len(cipherText)


    # code to find new length of cipher text after removing spaces and cipher text
    new_cipherlen =0;
    for i in range(cipherlen):
        if(cipherText[i].isalpha()):
            new_cipherlen+=1

    #code to return distinct letter count in cipher text
    #and sort it in alphabetical order
    letter_counts = assigment3_count_letters_in_cipherText(cipherText)
    print("Letter counts, arranged in alphabetical order:")
    for letter, count in sorted(letter_counts.items()):
        print(f"{letter} --> {count}")

    #code to return Frequency Distribution of letters in cipher text
    letter_frequency = assigment3_get_letter_frequency_distribution(cipherText)
    print("Frequency Distribution of letters in cipher text")
    for c in range(26):
        l = chr(ord('a')+c)
        if(letter_frequency.get(l,0))>0:
            p = (letter_frequency.get(l,0)/new_cipherlen * 100)
            print(f"{l} --> {p:.4f}")

    # code to invoke function to get permuted text
    permutedtext = assigment3_get_permuted_text(cipherText,cipherlen)

    # code to invoke char_replacement_rule function
    #to replace each letter with another as mentioned in rules
    rules = { "Ow": "O","Aq": "A","Ta": "T","Sl": "S","Ph": "P","Ev": "E","Hf": "H","Gg": "G","dd": "d","Rn": "R","Km": "K","Wr": "W",
            "Dp": "D","Ic": "I","yy": "y","jj": "j", "Fs": "F","Ce": "C","tt": "t","xx": "x","ii": "i","uu": "u","kk": "k","bb": "b"
    }

    permutedtext = assigment3_char_replacement_rule(permutedtext,rules)


    #code to replace each letter with another as mentioned in rules 1
    rules1 = { "jB","tL","xY","iQ","dU","yN","lI","bV","kJ","uM" }
    for rule1 in rules1:
        permutedtext = permutedtext.replace(rule1[0],rule1[1])


    permutedtext = permutedtext.lower()
    permutedtext +="\n"
    print("\n---permuted Text---")
    print(f"{permutedtext}")
    #code to reoder the words
    decryptedText = assigment3_re_order_letters(cipherText,permutedtext)

    print("\n---Decrypted Text---")
    print(f"{decryptedText}")


if __name__ == "__main__":
  main()


```