

16. Write a python program that can perform a letter frequency attack on any monoalphabetic substitution cipher without human intervention. Your software should produce possible plaintexts in rough order of likelihood. It would be good if your user interface allowed the user to specify “give me the top 10 possible plaintexts.”

# Simple monoalphabetic substitution cipher frequency attack

english = "ETAOINSHRDLCUMWFGYPBVKJXQ"

**Code:**

def score(t):

t = t.upper()

return sum(t.count(c) \* (26 - i) for i, c in enumerate(english))

cipher = input("Enter ciphertext: ")

top\_n = int(input("Top N plaintexts? "))

# Count ciphertext letter frequency

freq = {}

for c in cipher.upper():

if c.isalpha():

freq[c] = freq.get(c, 0) + 1

# Cipher letters sorted by frequency

cipher\_order = "".join(sorted(freq, key=freq.get, reverse=True))

# Build simple substitution map

mapping = {cipher\_order[i]: english[i] for i in range(len(cipher\_order))}

# Create plaintext

plain = ""

for ch in cipher:

if ch.upper() in mapping:

p = mapping[ch.upper()]

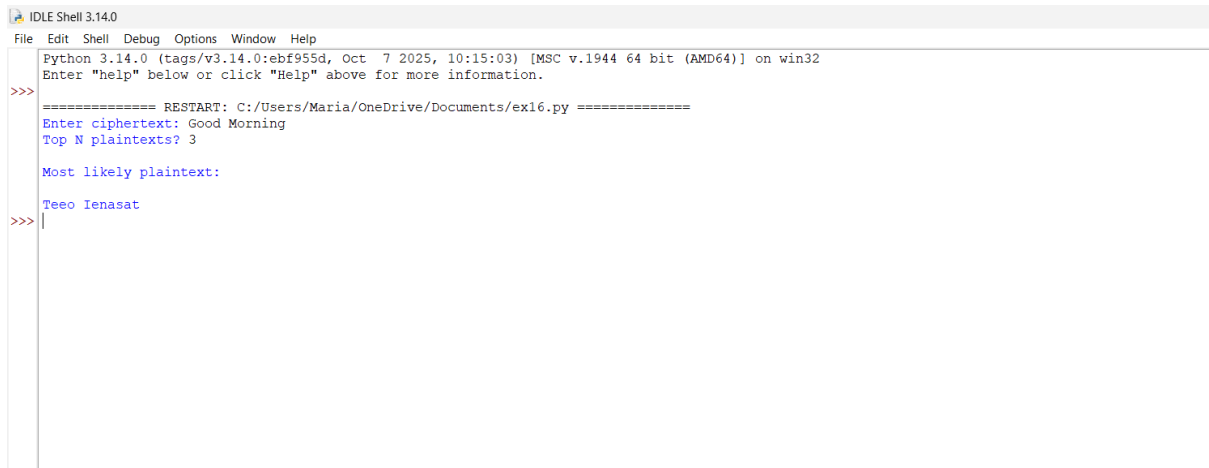
plain += p.lower() if ch.islower() else p

else:

```
plain += ch
```

```
print("\nMost likely plaintext:\n")
```

```
print(plain)
```



```
IDLE Shell 3.14.0
File Edit Shell Debug Options Window Help
Python 3.14.0 (tags/v3.14.0:ebf955d, Oct 7 2025, 10:15:03) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
===== RESTART: C:/Users/Maria/OneDrive/Documents/ex16.py =====
Enter ciphertext: Good Morning
Top N plaintexts? 3

Most likely plaintext:
Teeo Ienasat
>>> |
```