CS4023 Artificial Intelligence

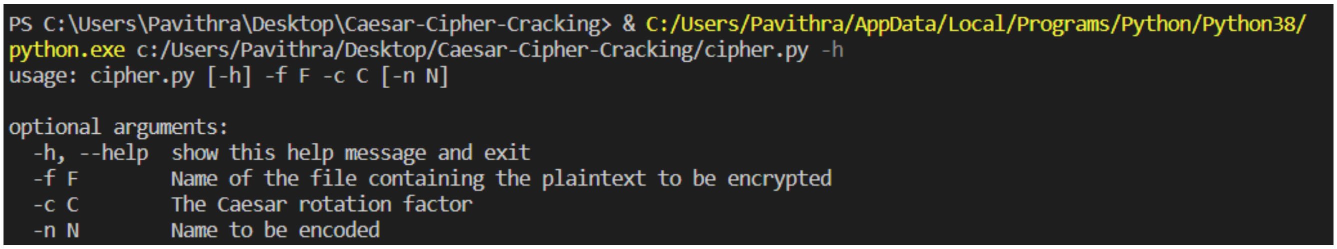
Name: Pavithra Rajan

Roll Number: B190632CS

1. Heuristics to solve the Caesar Cipher

- Utilise the **frequency distribution** of each alphabet in the English language and calculate the score obtained for each alphabet in the ciphertext and each iteration of all possibilities. The iteration with the **maximum** score is chosen.
- Use the **bigram frequency** distribution of the English language. A bigram is a pair of letters. 'Th' is the most common bigram. Similar to the alphabet frequency heuristic, we can compute the score and choose the iteration with the **maximum** score.
- As there are only limited alphabets in the English language (26 + 1 for spaces in sentences), we can try out all the possibilities and choose the one that resembles a valid text.

The `cipher.py` consists of the implementation of the aforementioned heuristics. The help menu can be obtained by running the `python` file with `-h` flag.

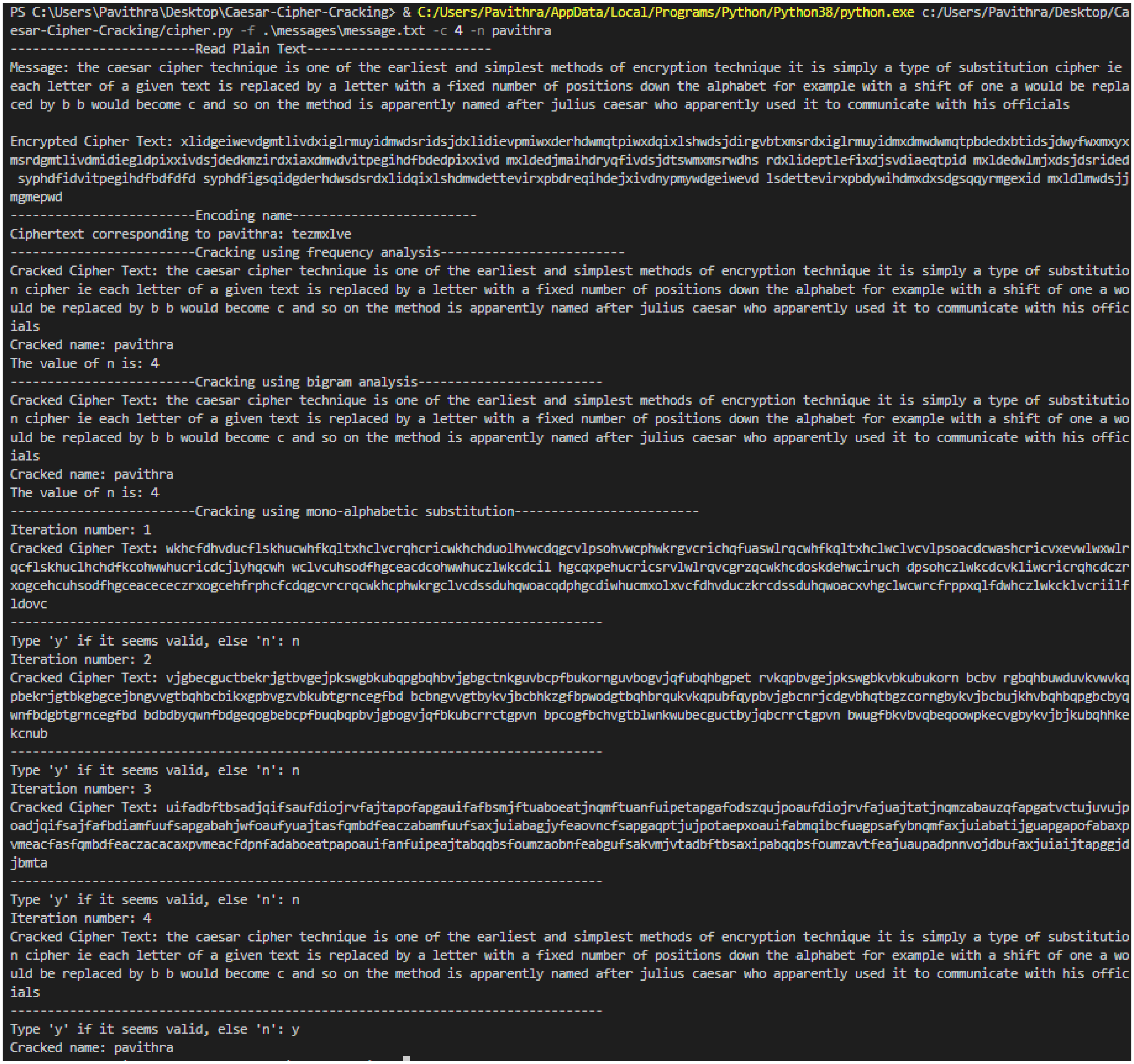


```
PS C:\Users\Pavithra\Desktop\Caesar-Cipher-Cracking> & C:/Users/Pavithra/AppData/Local/Programs/Python/Python38/
python.exe c:/Users/Pavithra/Desktop/Caesar-Cipher-Cracking/cipher.py -h
usage: cipher.py [-h] -f F -c C [-n N]

optional arguments:
  -h, --help  show this help message and exit
  -f F        Name of the file containing the plaintext to be encrypted
  -c C        The Caesar rotation factor
  -n N        Name to be encoded
```

While running the python file, provide the path to the message to be encrypted via the `-f` flag, the rotation factor with the `-n` flag, and the name to be encrypted following the `-p` flag.

I will provide a rotation factor of 4, and my name 'pavithra'.

Note: In order to correctly decrypt the name via frequency-based heuristics, it is necessary to have a sufficiently long text. To do this, I will concatenate a message along with the name and then extract it in the end after decryption.



```
PS C:\Users\Pavithra\Desktop\Caesar-Cipher-Cracking> & C:/Users/Pavithra/AppData/Local/Programs/Python/Python38/python.exe c:/Users/Pavithra/Desktop/Ca
esar-Cipher-Cracking/cipher.py -f .\messages\message.txt -c 4 -n pavithra
-------------------------Read Plain Text-------------------------
Message: the caesar cipher technique is one of the earliest and simplest methods of encryption technique it is simply a type of substitution cipher ie
each letter of a given text is replaced by a letter with a fixed number of positions down the alphabet for example with a shift of one a would be repla
ced by b b would become c and so on the method is apparently named after julius caesar who apparently used it to communicate with his officials

Encrypted Cipher Text: xlidgeiwevdgmtlivdxiglrmuyidmwdsridsjdxlidievpmiwxderhdwmqtpiwxdqixlshwdsjdirgvbtxmsrdxiglrmuyidmxdmwdwmqtpbdedxbtidsjdwyfwxmxyx
msrdgmtlivdmidiegldpixxivdsjdedkmzirdxiaxdmwdvitpegihdfbdedpixxivd mxldedjmaihdryqfivdsjdtswmxmsrwdhs rdxlideptlefixdjsvdiaeqtpid mxldedwlmjxdsjdsrided
 syphdfidvitpegihdfbdfdfd syphdfigsqidgderhdwsdsrdxlidqixlshdmwdettevirxpbdreqihdejxivdnypmywdgeiwevd lsdettevirxpbdywihdmxdxsdgsqqyrmgexid mxldlmwdsjj
mgmepwd
-------------------------Encoding name-------------------------
Ciphertext corresponding to pavithra: tezmxlve
-------------------------Cracking using frequency analysis-------------------------
Cracked Cipher Text: the caesar cipher technique is one of the earliest and simplest methods of encryption technique it is simply a type of substitutio
n cipher ie each letter of a given text is replaced by a letter with a fixed number of positions down the alphabet for example with a shift of one a wo
uld be replaced by b b would become c and so on the method is apparently named after julius caesar who apparently used it to communicate with his offic
ials
Cracked name: pavithra
The value of n is: 4
-------------------------Cracking using bigram analysis-------------------------
Cracked Cipher Text: the caesar cipher technique is one of the earliest and simplest methods of encryption technique it is simply a type of substitutio
n cipher ie each letter of a given text is replaced by a letter with a fixed number of positions down the alphabet for example with a shift of one a wo
uld be replaced by b b would become c and so on the method is apparently named after julius caesar who apparently used it to communicate with his offic
ials
Cracked name: pavithra
The value of n is: 4
-------------------------Cracking using mono-alphabetic substitution-------------------------
Iteration number: 1
Cracked Cipher Text: wkhcfdhvducflskhucwhfkqltxhclvcrqhcricwkhchduolhvwcdqgcvlpsohvwcphwkrgvcrichqfuaswlrqcwhfkqltxhclwclvcvlpsoacdcwashcricvxevwlwxwlr
qcflskhuclhchdfkcohwwhucricdcjlyhqcwh wclvcuhsodfhgceacdcohwwhuczlwkcdcil hgcqxpehucricsrvlwlrqvcgrzqcwkhcdoskdehwciruch dpsohczlwkcdcvkliwcricrqhcdczr
xogcehcuhsodfhgceacececzrxogcehfrphcfcdqgcvrcrqcwkhcphwkrgclvcdssduhqwoacqdphgcdiwhucmxolxvcfdhvduczkrcdssduhqwoacxvhgclwcwrcfrppxqlfdwhczlwkcklvcriilf
ldovc
--------------------------------------------------------------------------------
Type 'y' if it seems valid, else 'n': n
Iteration number: 2
Cracked Cipher Text: vjgbecguctbekrjgtbvgejpkswgbkubqpgbqhbvjgbgctnkguvbcpfbukornguvbogvjqfubqhbgpet rvkqpbvgejpkswgbkvbkubukorn bcbv rgbqhbuwduvkvwvkq
pbekrjgtbkgbgcejbngvvgtbqhbcbikxgpbvgzvbkubtgrncegfbd bcbngvvgtbykvjbcbhkzgfbpwodgtbqhbrqukvkqpubfqypbvjgbcnrjcdgvbhqtbgzcorngbykvjbcbujkhvbqhbqpgbcbyq
wnfbdgbtgrncegfbd bdbdbyqwnfbdgeqogbebcpfbuqbqpbvjgbogvjqfbkubcrrctgpvn bpcogfbchvgtblwnkwubecguctbyjqbcrrctgpvn bwugfbkvbvqbeqoowpkecvgbykvjbjkubqhhke
kcnub
--------------------------------------------------------------------------------
Type 'y' if it seems valid, else 'n': n
Iteration number: 3
Cracked Cipher Text: uifadbftbsadjqifsaufdiojrvfajtapofapgauifafbsmjftuaboeatjnqmftuanfuipetapgafodszqujpoaufdiojrvfajuajtatjnqmzabauzqfapgatvctujuvujp
oadjqifsajfafbdiamfuufsapgabahjwfoaufyuajtasfqmbdfeaczabamfuufsaxjuiabagjyfeaovncfsapgaqptjujpotaepxoauifabmqibcfuagpsafybnqmfaxjuiabatijguapgapofabaxp
vmeacfasfqmbdfeaczacacaxpvmeacfdpnfadaboeatpapoauifanfuipeajtabqqbsfoumzaobnfeabgufsakvmjvtadbftbsaxipabqqbsfoumzavtfeajuaupadpnnvojdbufaxjuiaijtapggjd
jbmta
--------------------------------------------------------------------------------
Type 'y' if it seems valid, else 'n': n
Iteration number: 4
Cracked Cipher Text: the caesar cipher technique is one of the earliest and simplest methods of encryption technique it is simply a type of substitutio
n cipher ie each letter of a given text is replaced by a letter with a fixed number of positions down the alphabet for example with a shift of one a wo
uld be replaced by b b would become c and so on the method is apparently named after julius caesar who apparently used it to communicate with his offic
ials
--------------------------------------------------------------------------------
Type 'y' if it seems valid, else 'n': y
Cracked name: pavithra
```

The encryption corresponding to ‘pavithra’ is: `tezmxlve`. It has been successfully decrypted as well.

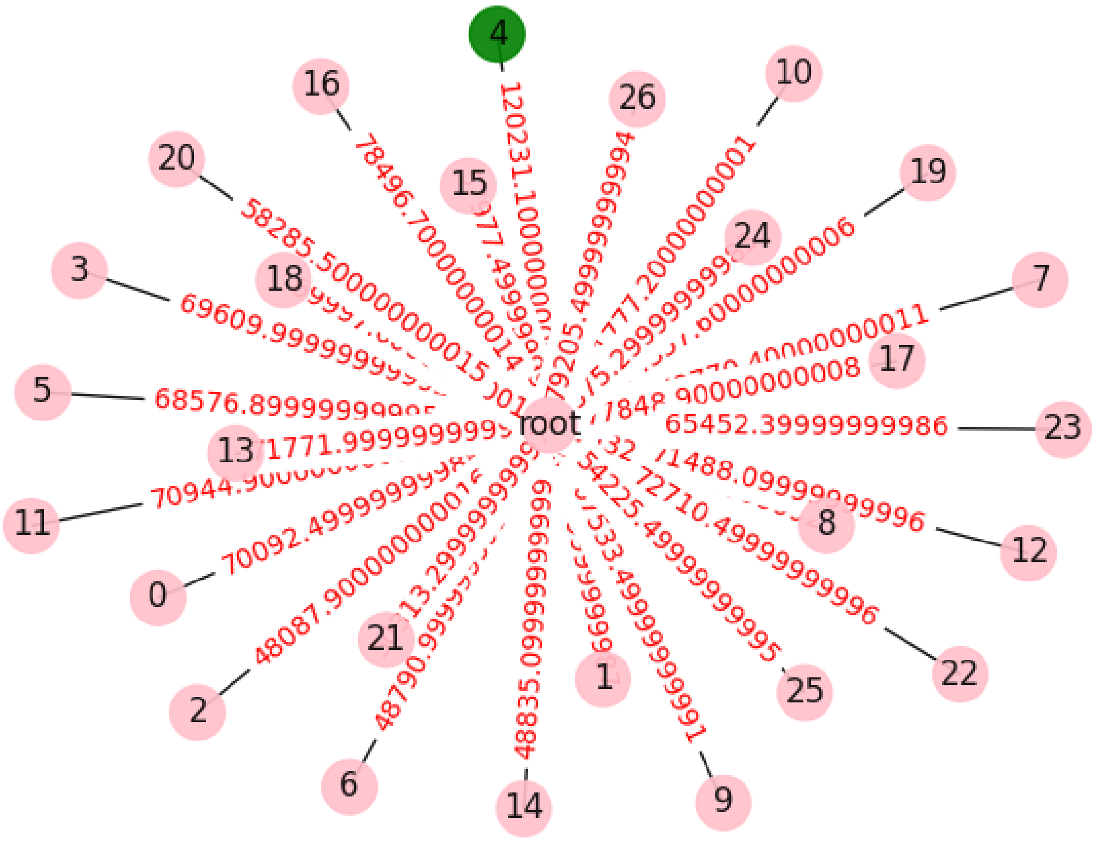
In the **mono-alphabetic substitution** method, I will check until I observe that the cracked text resembles a valid English text.

As we choose the best scores obtained across all iterations for the alphabet frequency and bigram frequency approach, they are **greedy methods.**

Along with the decryption of the ciphertexts, the **state-space graphs** are plotted using the `networkx` library in python.

- **Alphabet frequency method**

As we can see from the graph, the maximum score is when the rotation factor is 4.

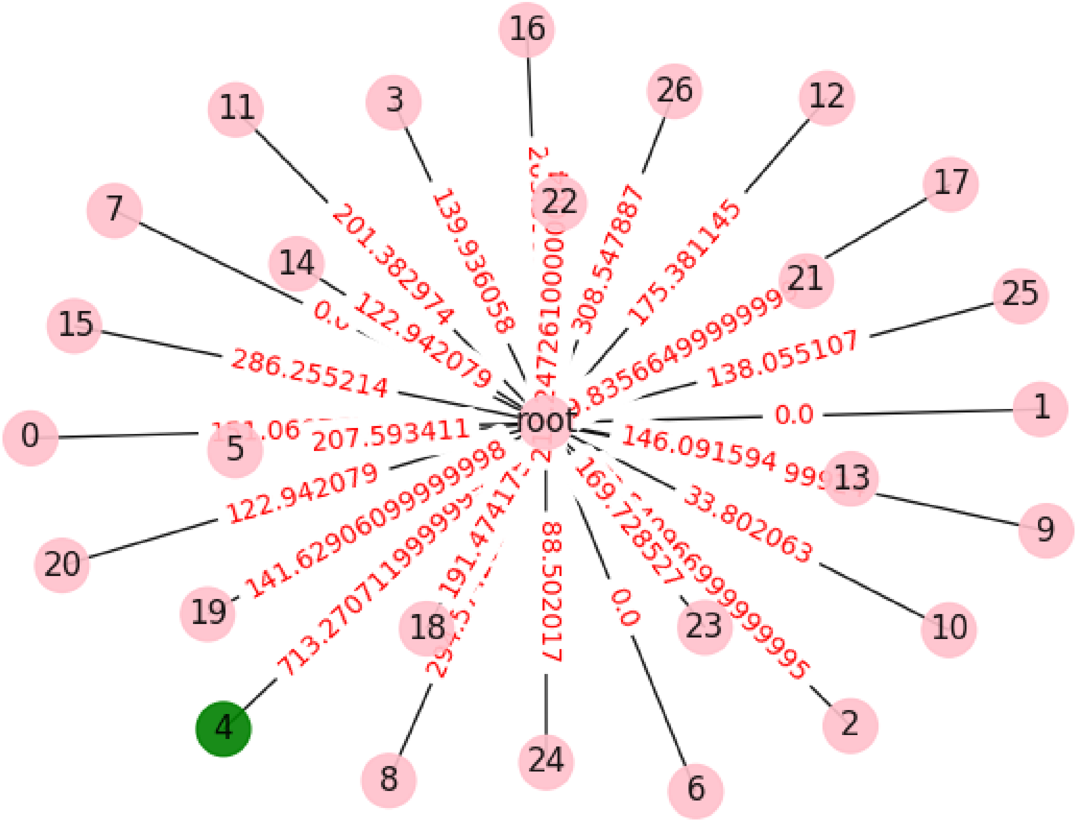


- **Bigram frequency**

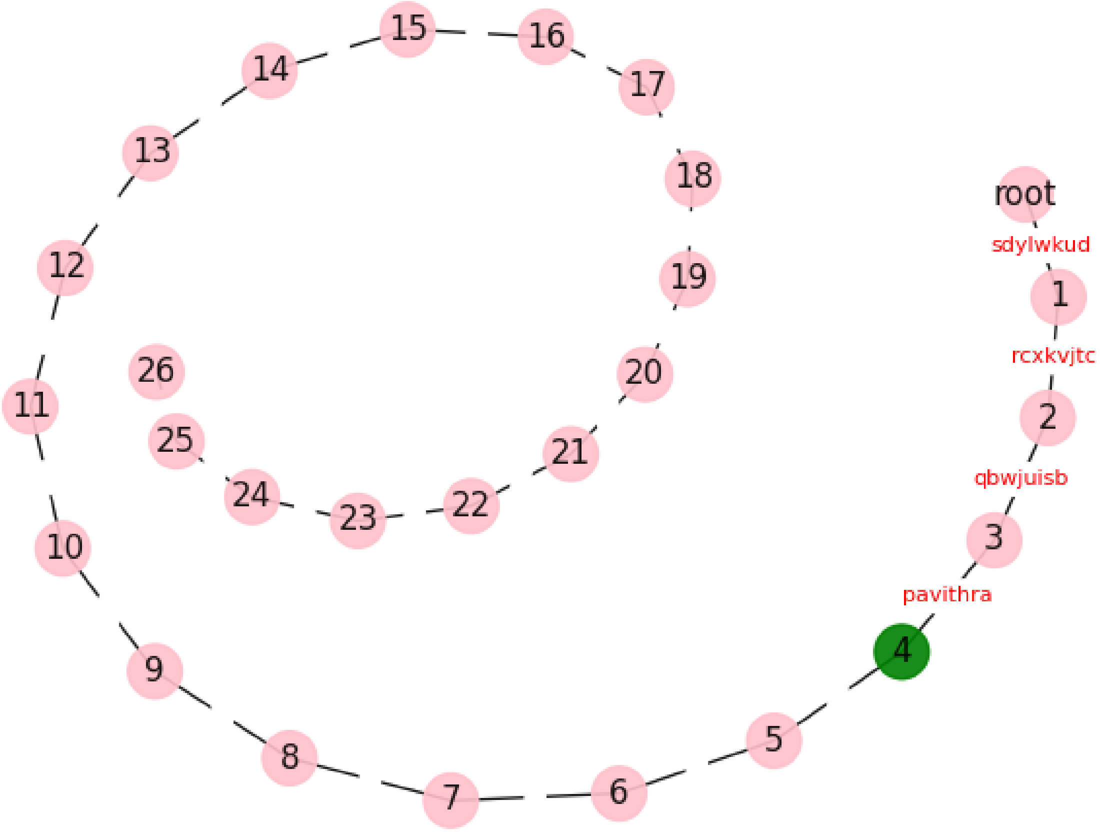
As we can see from the graph, the maximum score is when the rotation factor is 4.

- **Mono-alphabetic substitution**

Here, we can see all the possible decryptions of my name across each rotation factor till 4.



root
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
201.382974
139.936058
308.547887
175.381145
122.942079
286.255214
207.593411
138.055107
0.0
146.091594
33.802063
122.942079
141.62906099999998
713.27071199999
191.47417
88.502017
0.0
169.728527
9999999995

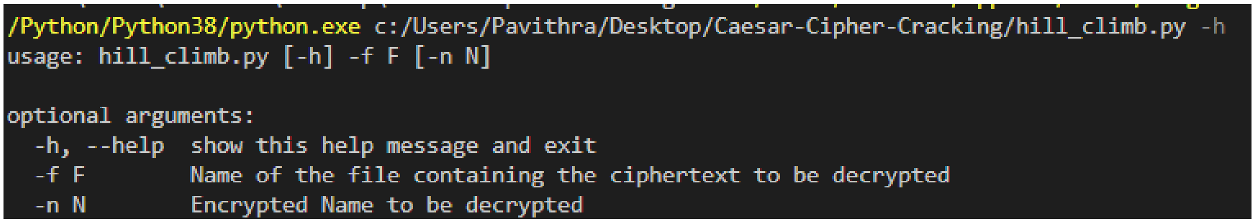


root
sdylwkud
1
rcxkvjtc
2
qbwjuisb
3
pavithra
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26

2. Hill Climbing

In order to show hill climbing, I will use the quadgram frequencies.

The `hill_climb.py` consists of this implementation. The help menu can be obtained by running the `python` file with `-h` flag.



```
/Python/Python38/python.exe c:/Users/Pavithra/Desktop/Caesar-Cipher-Cracking/hill_climb.py -h
usage: hill_climb.py [-h] -f F [-n N]

optional arguments:
  -h, --help  show this help message and exit
  -f F        Name of the file containing the ciphertext to be decrypted
  -n N        Encrypted Name to be decrypted
```

While running the python file, provide the path to the ciphertext to be decrypted via the `-f` flag and the encrypted name with the `-n` flag.

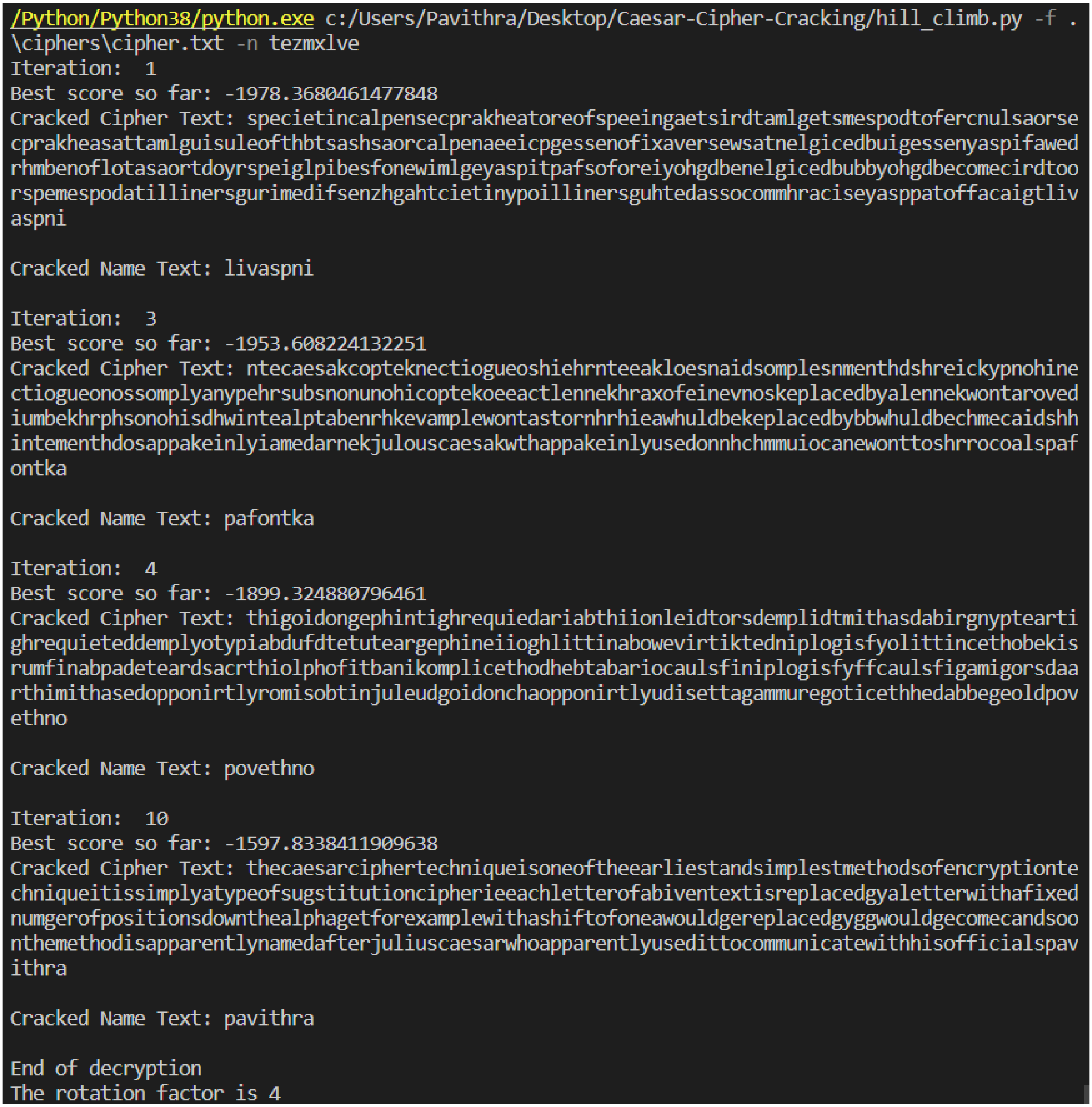
I will run the python file by giving the path to the file containing the encrypted text along with my name encrypted (tezmxlve). Hit Ctrl + C if the decrypted text looks valid.

Here the scores for each iteration will be computed till a **local maximum** is reached and shown to the user. If the text resembles a valid English text, then we can stop.

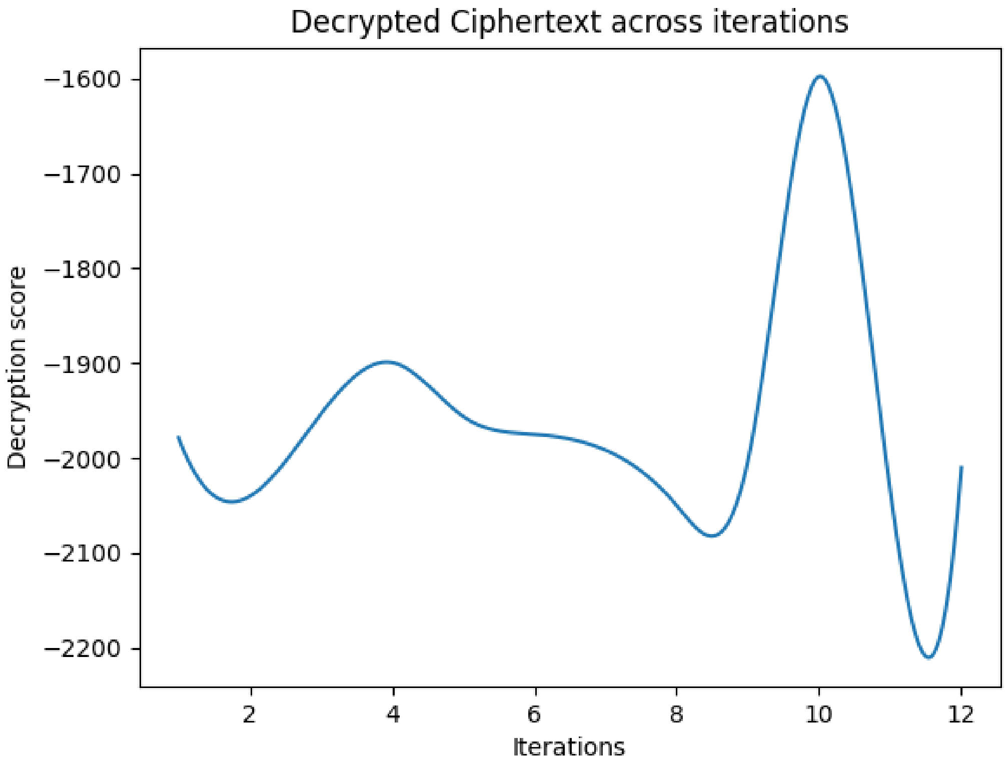
It takes 10 iterations for the hill climbing method to correctly decrypt my name. It also tells the rotation factor as well. We can see below that the scores progressively increase (becomes less negative). The graph depicting the scores across each iteration is also generated. The last picture shows the possible decryptions of my name. The yellow nodes depict local maximas and green depicts the final goal node.

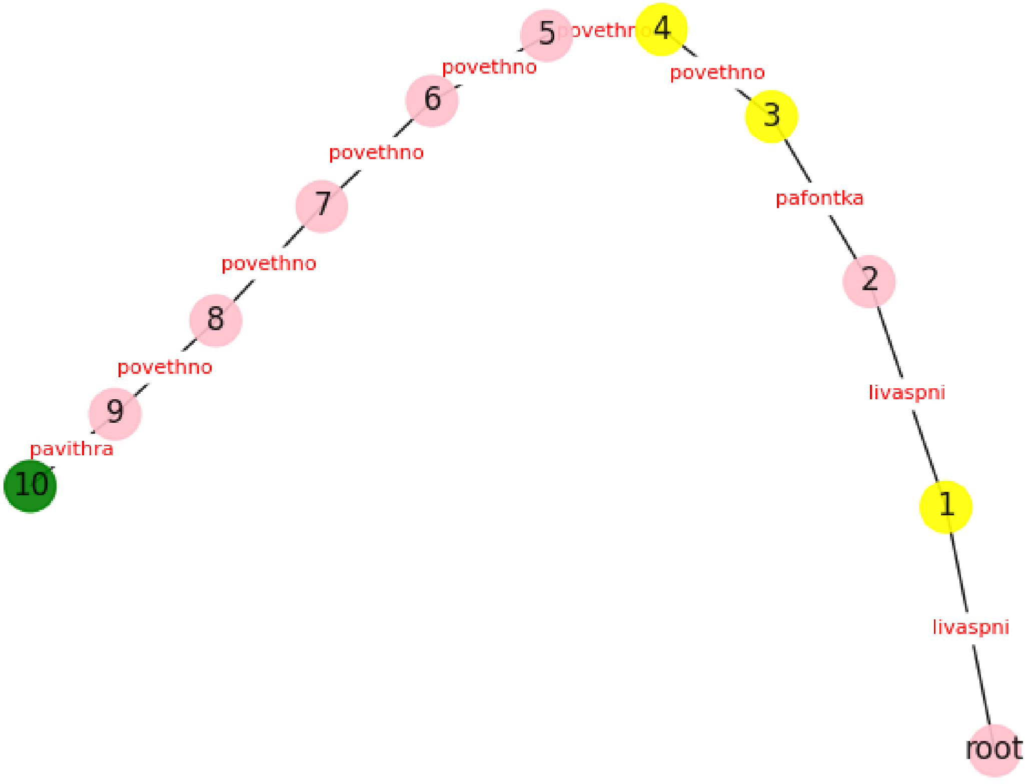
3. Conclusion

- I observed that the Greedy approach performed better than the Hill Climbing method. The state space graph had extended upto only one level of depth and choosing the maximum score yielded the correct result.
- In the case of Hill Climbing, it was noticed that sometimes the process got stuck at local maximas and required more number of iterations to correctly decrypt the ciphertext. Hence, this approach took greater execution time.



```
/Python/Python38/python.exe c:/Users/Pavithra/Desktop/Caesar-Cipher-Cracking/hill_climb.py -f .
\ciphers\cipher.txt -n tezmxlve
Iteration:  1
Best score so far: -1978.3680461477848
Cracked Cipher Text: specietincalpensecprakheatoreofspeeingaetsirdtamlgetsmespodtofercnulsaorse
cprakheasattamlguisuleofthbtsashsaorcalpenaeeicpgessenofixaversewsatnelgicedbuigessenyaspifawed
rhmbenoflotasaortdoyrspeiglpibesfonewimlgeyaspitpafsoforeiyohgdbenelgicedbubbyohgdbecomecirdtoo
rspemespodatillinersgurimedifsenzhgahtcietinypoillinersguhtedassocommhraciseyasppatoffacaigtliv
aspni

Cracked Name Text: livaspni

Iteration:  3
Best score so far: -1953.608224132251
Cracked Cipher Text: ntecaesakcopteknectiogueoshiehrnteeakloesnaidsomplesnmenthdshreickypnohine
ctiogueonossomplyanypehrsubsnonunohicoptekoeeactlennekhraxofeinevnoskeplacedbyalennekwontaroved
iumbekhrphsonohisdhwintealptabenrhkevamplewontastornhrhieawhuldbekeplacedbybbwhuldbechmecaidshh
intementhdosappakeinlyiamedarnekjulouscaesakwthappakeinlyusedonnhchmmuiocanewonttoshrrocoalspaf
ontka

Cracked Name Text: pafontka

Iteration:  4
Best score so far: -1899.324880796461
Cracked Cipher Text: thigoidongephintighrequiedariabthiionleidtorsdemplidtmithasdabirgnypteartі
ghrequieteddemplyotypiabdufdtetuteargephineiioghlittinabowevirtiktedniplogisfyolittincethobekis
rumfinabpadeteardsacrthiolphofitbanikomplicethodhebtabariocaulsfiniplogisfyffcaulsfigamigorsdaa
rthimithasedopponirtlyromisobtinjuleudgoidonchaopponirtlyudisettagammuregoticethhedabbegeoldpov
ethno

Cracked Name Text: povethno

Iteration:  10
Best score so far: -1597.8338411909638
Cracked Cipher Text: thecaesarciphertechniqueisoneoftheearliestandsimplestmethodsofencryptionte
chniqueitissimplyatypeofsugstitutioncipherieeachletterofabiventextisreplacedgyaletterwithafixed
numgerofpositionsdownthealphagetforexamplewithashiftofoneawouldgereplacedgyggwouldgecomecandsoo
nthemethodisapparentlynamedafterjuliuscaesarwhoapparentlyusedittocommunicatewithhisofficialspav
ithra

Cracked Name Text: pavithra

End of decryption
The rotation factor is 4
```


Decrypted Ciphertext across iterations
Decryption score
−1600
−1700
−1800
−1900
−2000
−2100
−2200
Iterations
2
4
6
8
10
12


5
povethno
4
povethno
6
povethno
3
povethno
7
pafontka
povethno
2
8
povethno
livaspni
9
pavithra
10
1
livaspni
root