

ΘΕΩΡΙΑ ΠΛΗΡΟΦΟΡΙΩΝ ΚΑΙ ΚΩΔΙΚΩΝ

Μέλη Ομάδας

- Νικόλας Πατέρας Π17172
- Ανδρέας Θεοδωρίδης Π17164
- Βασίλειος Παπαχαραλάμπους Π17168

? Εκφώνηση Εργασίας

Σκοπός των εργασιών είναι η αποστολή και λήψη μιας εικόνας από έναν ένα ζεύγος πομπού-δέκτη. Ο πομπός παίρνει από τον χρήστη <mark>σαν είσοδο</mark> ένα <mark>κείμενο</mark> από αρχείο κειμένου (απλό ASCII text αρχείο), <mark>το μήκος του κώδικα</mark> και ότι άλλες παραμέτρους χρειάζεται για να τον δημιουργήσει **αλλά και μία τιμή** που καθορίζει την <mark>ένταση του θορύβου</mark>. Λαμβάνοντας το κείμενο, <mark>το πρόγραμμά σας **συμπιέζει**</mark> με τον αντίστοιχο κώδικα <mark>το κείμενο</mark> (χρησιμοποιήστε τα **στατιστικά** των γραμμάτων όπου χρειάζεται). Στην συνέχεια, <mark>το συμπιεσμένο</mark> string το κωδικοποιείτε με τον αντίστοιχο κώδικα και του προσθέτετε το αντίστοιχο θόρυβο (πχ ένταση **5** σημαίνει ότι αλλάζετε μέχρι **5 <mark>bit** στην κάθε λέξη που **αποστέλλετε**) Το</mark> αποτέλεσμα είναι ένα string το οποίο το κωδικοποιείτε σε base64 και στέλνετε στον παραλήπτη ένα **json** το οποίο περιέχει το όνομα του αλγόριθμου συμπίεσης, το είδος του κώδικα, **το μήκος του κώδικα** (και ότι άλλες παραμέτρους χρειάζεστε) και το κωδικοποιημένο string. π.χ.

```
"compression algorithm":"LZ78",
"code":{
 "name":"linear",
 "P":[[1,0,0...]],
```

Μόλις ο πομπός αποστείλει το μήνυμά του, εμφανίζει μήνυμα στο χρήστη με στατιστικά:

Αρχικό μέγεθος αρχείου, εντροπία, τελικό μέγεθος αρχείου, πόσα bits προστέθηκαν.



🛕 Ζητούμενα

Γραμμικός Κώδικας και Lempel-Ziv 78

ΜΕπεξήγηση Κώδικα/Λύσης

Αρχικά, ανοίγουμε δυο terminals, στο πρώτο εκτελούμε την εντολή **py Dektis.py** και στο δεύτερο **py Pompos.py** έτσι ώστε να ενωθεί ο server με τον client. Στην

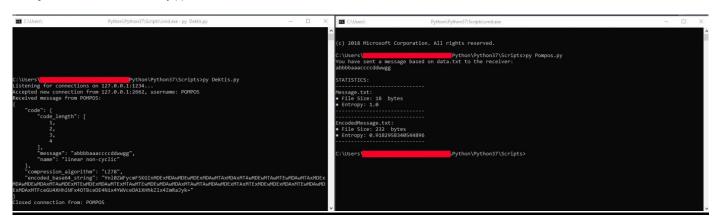
συνέχεια στο αρχείο **Pompos.py** καλείται η συνάρτηση **getMessage()** όπου παίρνει το κείμενο που βρίσκεται στο αρχείο **Message.txt** και στην συνέχεια γίνεται η συμπίεση με τον αλγόριθμο **LZ78** όπου θα μας δώσει ένα <u>δυαδικό αριθμό</u> σαν αποτέλεσμα.

Μετά γίνεται η κωδικοποίηση με γραμμικό κώδικα (reedsolo.py).

Εφόσον ολοκληρωθεί αυτό το κομμάτι, θα προστεθούν τα αντίστοιχα **bits** στο αποτέλεσμα (**NoiseCode.py**).

Επίσης, γίνεται κωδικοποίηση του αποτελέσματος σε **base64**, αποθηκεύεται το αποτέλεσμα στο αρχείο **EncodedMessage.txt**, αποστέλλεται στον <u>Δέκτη</u> και εμφανίζεται σε μορφή **JSON** με όλα τα σχετικά που ζητήθηκαν.

Τέλος, εμφανίζεται στον <u>Πομπό</u> το μέγεθος του αρχείου **Message.txt** (Αρχικό μέγεθος αρχείου) και **EncodedMessage.txt** (Τελικό μέγεθος αρχείου) και η **εντροπία** των δυο αρχείων.



🚺 Πληροφορίες - Πηγές

- Γλώσσα Προγραμματισμού
 - Python 3.7.3 (Ο κώδικας δεν λειτουργεί σε προηγούμενες εκδόσεις τις python)
- Chat room app για πομπό-δέκτη

https://www.youtube.com/watch?v=CV7 stUWvBQ

https://www.youtube.com/watch?v=ytu2yV3Gn1I

https://pythonprogramming.net/server-chatroom-sockets-tutorial-python-3/

https://pythonprogramming.net/client-chatroom-sockets-tutorial-python-

3/?completed=/server-chatroom-sockets-tutorial-python-3/

- Βιβλιοθήκη για τον υπολογισμό της εντροπίας https://rosettacode.org/wiki/Entropy#Python
- Βοήθεια για το πως θα πάρουμε τα δεδομένα από το αρχείο https://www.w3schools.com/python/python file open.asp
- Μετατροπή String σε base64
 https://stackoverflow.com/questions/13261802/converting-a-string-to-and-from-base-64/13267801
- Αλγόριθμος LZ78
 https://github.com/N03/LZ78.py/blob/master/LZ78.py
- Γραμμικός Κώδικας
 https://github.com/tomerfiliba/reedsolomon/blob/master/reedsolo.py
 https://www.sciencedirect.com/topics/computer-science/generator-polynomial
 https://www.python-course.eu/polynomial class in python.php
 https://www.sanfoundry.com/python-program-compute-polynomial-equation-coefficients-list/
- Μέγεθος αρχείου
 https://stackoverflow.com/questions/6591931/getting-file-size-in-python



