ΘΕΩΡΙΑ ΠΛΗΡΟΦΟΡΙΩΝ ΚΑΙ ΚΩΔΙΚΩΝ

Π18126 ΑΘΑΝΑΣΙΟΣ ΠΕΤΡΟΠΟΥΛΟΣ
Π18139 ΦΑΙΗ ΣΙΟΣΙΟ

ΓΕΝΙΚΗ ΙΔΕΑ ΕΡΓΑΣΙΑΣ 1

Οι κώδικες C1,C2 είναι και οι δύο γραμμικοί διάστασης k και μήκους n οπότε έχουν την μορφή: $C=[I_k ; P]$.

Όπου I διαγώνιος πίνακας δίαστασης $k*k$ και
P τυχαίος πίνακας διαστάσεων $(n-k)*k$.

Για την υλοποίηση της εργασίας έχει χρησιμοποιηθεί η γλώσσα python3 και το πακέτο sagemath.Υπάρχουν 3 αρχεία το encode.py,decode.py και main.py.

ΕΠΕΞΗΓΗΣΗ ΚΩΔΙΚΑ

Στο encode υλοποιούνται οι συναρτήσεις οι οποίες καλούνται για την κωδικοποίηση του μηνύματος.

Αρχικά υλοποιείται η συνάρτηση generateCode που παίρνει σαν όρισματα το n(μήκος) και το k(διάσταση),φτιάχνει τον πίνακα γεννήτρια και χρησιμοποιόντας την μέθοδο LinearCode του πακέτου sagemath,υλοποιεί τον γραμμικό κώδικα.

Έπειτα υλοποιείται η συνάρτηση messageConverter η οποία τροποποιεί το μήνυμα με τον εξής τρόπο:

1. Μετατροπή του κάθε χαρακτήρα σε δυαδικού.
2. Αφαίρεση του προθέματος.
3. Πρόσθεση αρκετών μηδενικών στην αρχή ώστε κάθε χαρακτήρας να έχει μήκος 8.

Η συνάρτηση gMaker επιστρέφει τον πίνακα $S*G*P$ Παίρνει σαν είσοδο τον πίνακα γεννήτρια του κώδικα.Έχοντας τις διαστάσεις του G μπορεί να δημιουργήσει τους πίνακες S και P σε κατάλληλες διαστάσεις.Ο πίνακας P υλοποιείται μεταθέτοντας τις σειρές του εκάστοτε μοναδιαίου.Ο πίνακας S προσθέτοντας γραμμές του μοναδιαίου ώστε κάθε φορά να δημιουργείται διαφορετικός πίνακας.Αρχικά γίνεται ο πολλαπλασιασμός $S*G$ και έπειτα το γινόμενο αυτό με τον πίνακα P.

Η συνάρτηση cMaker υλοποιεί την πράξη $c*S*G*P$. Αυτό επιτυγχάνεται χωρίζοντας το μήνυμα σε κατάλληλα τμήματα ώστε να μπορεί να υλοποιηθεί ο πολλαπλασιασμός.

Η συνάρτηση prepareForEncode προσθέτει έξτρα bits στο τέλος του μηνύματος,ώστε το μήνυμα να χωριστεί σε κομμάτια τα οποία στο σύνολο τους να μπορούν να κωδικοποιηθούν απο τον κώδικα μήκους n.

Η συνάρτηση errorEncode παίρνει ως είσοδο την ελάχιστη απόσταση του κώδικα και το μήνυμα και προσθέτει ένα διάνυσμα λάθους που δημιουργεί λάθη σε βαθμό που ο κώδικας μπορεί να τα αναγνωρίσει.

Η συνάρτηση encode παίρνει ως είσοδο το μήνυμα την διάσταση του κώδικα και το μήκος του μηνύματος.Επιστρέφει το κωδικοποίημένο μήνυμα και στο τέλος προσθέτει το μήκος του μηνύματος σε δεκαεξαδική μορφή.

Στο αρχείο decode.py υλοποιείται η αποκωδικοποίηση με την χρήση της συνάρτησης decode.Η οποία παίρνει σαν είσοδο τον κώδικα,το μήνυμα και το μήκος του καθώς και το μήκος της λέξης.Αρχικά το μήνυμα μετατρέπεται σε αριθμητική λίστα η οποία χωρίζεται σε μηνύματα n μεγέθους.Το σύστημα αφού δημιουργήσει ένα αντικείμενο decoder λαμβάνει ως είσοδο διανύσματα καταλλήλου μήκους και τα αποκωδικοποιεί.Έπειτα χρησιμοποιόντας το μήκος “πετάει” τα τελευταία bit τα οποία προστέθηκαν ώστε ο κώδικας να δουλέψει σωστά.Τέλος εκτυπώνει το μήνυμα και το επιστρέφει.

Στο αρχείο main.py εμφανίζονται τα κατάλληλα μηνύματα ώστε ο χρήστης να είσαγει τα μήκη και τις διαστάσεις των κωδίκων,καθώς και το μήνυμα που θα κωδικοποιηθεί.Σε όλη την διάρκεια της εκτέλεσης εκτυπώνει την πρόοδο και τα αποτελέσματα της κάθε ενέργειας.

ΠΑΡΑΔΕΙΓΜΑ ΕΚΤΕΛΕΣΗΣ

Το παράδειγμα εκτέλεσης κωδικοποιεί το μήνυμα Hello World με κώδικες C1,C2 μήκους 7 και 8 και διάστασης 4 και 5 αντίστοιχα.

```
=====FIRST CODE=====
-->please enter length of Code (n)
7
-->please enter dimension of Code (k)
4
=====SECOND CODE=====
-->please enter length of Code (n)
8
-->please enter dimension of Code (k)
5
=====MESSAGE=====
-->please enter the message you want encoded
Hello World
```

Γεννήτριες πίνακες και ελάχιστες αποστάσεις

```
-->Generator matrix for code no 1
[1 0 0 0 0 1 0]
[0 1 0 0 0 1 0]
[0 0 1 0 1 0 0]
[0 0 0 1 0 1 0]
--Minimum distance for code no 1
2



-->Generator matrix for code no 2
[1 0 0 0 0 0 1 1]
[0 1 0 0 0 0 1 1]
[0 0 1 0 0 1 0 0]
[0 0 0 1 0 1 1 1]
[0 0 0 0 1 0 1 0]
--Minimum distance for code no 2
2
```

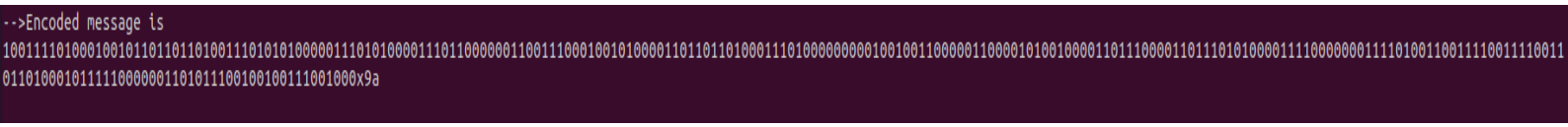
Μήνυμα σε δυαδική μορφή

```
01001000011001010110110001101100011011110010000001010111011011110111001001101100011001000
```

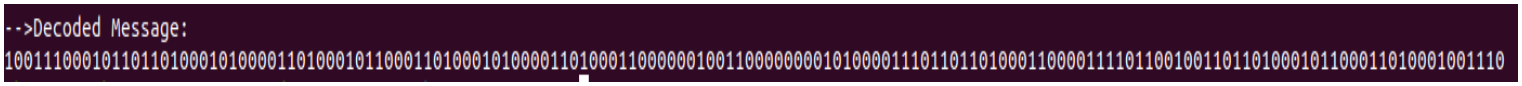
Πίνακες S,P και $S*G*P$

```
-->Array S(random invertible array)
[0 0 1 0]
[1 0 0 0]
[0 1 0 0]
[0 0 0 1]



-->Array P(random identity array permutation)
[1 0 0 1 0 0 0]
[0 1 0 0 0 0 0]
[0 0 1 0 1 1 0]
[0 0 1 1 0 0 0]
[0 0 0 0 0 0 0]
[0 0 0 0 1 1 0]
[0 0 0 0 0 0 1]



-->SxGxP
[0 0 1 0 1 1 0]
[1 0 0 1 1 1 0]
[0 1 0 0 1 1 0]
[0 0 1 1 1 1 0]
```

Κωδικοποιημένο μήνυμα



Αποκωδικοποιημένο μήνυμα



<u>ΣΗΜΕΙΩΣΕΙΣ</u>

Όλα οι πίνακες,τα διανύσματα και οι τιμές είναι πεδία Γκαλουά(GF(2)),επομένως όλες οι πράξεις γίνονται στον δυαδικό χώρο.Οι μέθοδοι matrix,vector,rows,columns προέρχονται απο το sagemath.