

Κ23γ: Ανάπτυξη Λογισμικού για Αλγοριθμικά Προβλήματα

Χειμερινό εξάμηνο 2018-19

1 η Προγραμματιστική Εργασία

Υλοποίηση δομής για την εύρεση κοντινών γειτόνων στη γλώσσα C/C++

Ο φάκελος της εργασίας περιλαμβάνει 9 αρχεία εκ των οποίων τα 5 είναι πηγαία, τα 2 είναι αρχεία-βιβλιοθήκες και το τελευταίο είναι το Makefile για τη διευκόλυνση της μεταγλώττισης του κώδικα.

Ως προς τα αρχεία, τα προγράμματα διαφέρουν μόνο στο αρχείο main. Το πρόβλημα LSH περιλαμβάνει το αρχείο main_lsh.cpp, ενώ ο υπερκύβος χρειάζεται το main_cube.cpp. Τα υπόλοιπα αρχεία είναι εξίσου απαραίτητα.

Στην υλοποίησή μου έχω δημιουργήσει 3 βασικές κλάσεις: Hashlist, Hashtable και Point, η κάθε μία από τις οποίες εξαρτάται από την επόμενη της στη δοσμένη σειρά. Ο υπερκύβος έχει προκύψει ως κλάση-παιδί της Hashtable ώστε να κληρονομήσει τα χαρακτηριστικά που χρειάζονται για τη δημιουργία των συναρτήσεων h() που χρειάζονται στη συνάρτηση κατακερματισμού.

Για τη μεταγλώττιση του κώδικα αρκεί η εντολή make all. Η εντολή εκτέλεσης των προγραμμάτων ακολουθεί τον κανόνα που δόθηκε στην εκφώνηση, οπότε αρκούν τα παρακάτω:

```
$/lsh -d <input file> -q <query file> -k <int> -L <int> -o <output file>  
$/cube -d <input file> -q <query file> -k <int> -M <int> -probes <int> -o <output file>
```

Ωστόσο, όλα τα flags είναι προαιρετικά.

Κατά τη διάρκεια της ανάπτυξης χρησιμοποιήθηκε το GitHub για version control της εργασίας. Το αντίστοιχο repository βρίσκεται στο σύνδεσμο:
<https://github.com/sanantoniochili/LSH-and-Hypercube.git>