ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
Εθνικόν και Καποδιστριακόν
Πανεπιστήμιον Αθηνών
ΙΔΡΥΘΕΝ ΤΟ 1837

Κατακερματισμός και αναζήτηση για χρονοσειρές στη C/C++

Ανάπτυξη Λογισμικού για Αλγοριθμικά Προβλήματα

2η Προγραμματιστική Εργασία

ΔΗΜΗΤΡΗΣ ΧΑΜΑΡΙΑΣ - 1115201600190

ΑΝΤΩΝΙΑ ΡΟΥΣΣΟΥ - 1115201600147

Χειμερινό εξάμηνο 2021-2022

Κατάλογος αρχείων

Δομή φακέλων / αρχείων

- Φάκελος **cluster** με τα αρχεία:
  - *cluster.conf* (Αρχείο ρύθμισης παραμέτρων)
  - *cluster.cpp* (Εκτέλεση αλγορίθμων για τη συσταδοποίηση χρονοσειρών)
  - *clusteringMethods.cpp* (Ορισμοί μεθόδων της κλάσης Cluster)
  - *clusteringMethods.hpp* (Ορισμός της κλάσης Cluster)
- Φάκελος **search** με τα αρχεία:
  - *search.cpp* (Εκτέλεση αλγορίθμων για την αναζήτηση πλησιέστερου γείτονα χρονοσειρών με τις μετρικές Discrete/Continuous Frechet)
  - *cubeSearch.cpp* (Ορισμός αλγορίθμων Hypercube)
  - *cubeSearch.hpp* (Δηλώσεις αλγορίθμων(συναρτήσεων) Hypercube)
  - *lshSearch.cpp* (Ορισμός αλγορίθμων LSH)
  - *lshSearch.hpp* (Δηλώσεις αλγορίθμων(συναρτήσεων) LSH)
- Φάκελος **datasets** με τα αρχεία (σύνολο δεδομένων και σύνολο αναζήτησης):
  - *nasd_input.csv*
  - *nasd_query.csv*
  - *nasdaq2015_2017.csv*
  - *nasdaq2017_LQ.csv*
- Φάκελος **utilities** με τα αρχεία:
  - *hash.cpp* (Ορισμός των μεθόδων HashTable)
  - *hash.hpp* (Ορισμός της κλάσης HashTable και της κλάσης Data)
  - *curve.cpp (Ορισμός της κλάσης Curve)*
  - *curve.hpp (Ορισμός των μεθόδων Curve)*
  - *grid.cpp (Ορισμός της κλάσης Grid)*
  - *grid.hpp (Ορισμός των μεθόδων Grid)*
  - *hypercube.cpp* (Ορισμός των μεθόδων Hypercube)
  - *hypercube.hpp* (Ορισμός της κλάσης Hypercube)
  - *metrics.cpp* (Ορισμός των μετρικών, ευκλείδεια απόσταση, manhattan, hamming και Discrete Frechet)
  - *metrics.hpp* (Δηλώσεις μετρικών)
  - *PriorityQueue.cpp* (Ορισμός των μεθόδων PriorityQueue)

- *PriorityQueue.hpp* (Ορισμός της κλάσης PriorityQueue)
- *utilities.cpp* (Ορισμός κοινών συναρτήσεων για parsing και διάβασμα/εκτύπωση αρχείων)
- *utilities.hpp* (Ορισμός του struct Neighbor και δηλώσεις κοινών συναρτήσεων)
- *completeBinaryTree.cpp* (Ορισμός των μεθόδων CompleteBinaryTree)
- *completeBinaryTree.hpp* (Ορισμός της κλάσης CompleteBinaryTree)

- Φάκελος **tests** με τα αρχεία (Unit Tests):
  - *testBinaryTree.cpp* (Έλεγχος της κατασκευής του complete binary tree από καμπύλες, πλήθος κόμβων)
  - *testDiscFrechet.cpp* (Έλεγχος της μετρικής discrete Frechet, απόσταση 2 καμπυλών)
  - *testHashTable.cpp* (Έλεγχος κατασκευής Hash table, πλήθος στοιχείων μετά από εισαγωγές)
  - *runTests.cpp* (Εκτέλεση tests)
  - *test.hpp* (Δηλώσεις test συναρτήσεων)

Περιγραφή κλάσεων / structs

Data

Αντιπροσωπεύει ένα διάνυσμα και έχει τις εξής πληροφορίες: το ίδιο το διάνυσμα ***vec***, το id του ***id,*** το cluster στο οποίο ανήκει ***cluster*** και την απόσταση του πλησιέστερου cluster ***minDist***.

HashTable

Η βασική δομή ενός hash table που περιέχει τις εξής πληροφορίες: το μέγεθος του πίνακα ***size***, τον ίδιο τον πίνακα ***table*** που υλοποιείται ως ένα array από λίστες(η κάθε λίστα περιέχει δείκτες σε Data), το πλήθος των στοιχείων του ***containedItems*** και μια λίστα από τις hash functions ***hashFunctions***.

Hypercube

Η δομή του υπερκύβου που κληρονομεί από την κλάση HashTable και περιέχει τις επιπλέον πληροφορίες: μια λίστα από unordered maps που περιέχουν για κάθε συνάρτηση h, ζευγάρια <τιμή h, 0 ή 1>.

Neighbor

Αναπαριστά έναν γείτονα (πλησιέστερο σημείο) και αποτελείται από ένα ***id*** και την απόστασή του από ένα query ***dist***.

Centroid

Αντιπροσωπεύει ένα κεντροειδές και αποτελείται από τις εξής πληροφορίες: το κεντροειδές ως διάνυσμα ***vec***, το άθροισμα των σημείων που περιέχει ***vecSum***, τα indexes των σημείων που περιέχει ***indexes*** και το μεσο silhouette κάθε σημείου ***silhouette***.

Cluster

Η βασική κλάση για τη συσταδοποίηση, η οποία αποτελείται από τις εξής πληροφορίες: το σύνολο των σημείων (vector) ***points***, το σύνολο των κεντροειδών ***centroids***, το πλήθος των clusters ***K***, την μέθοδο ανάθεσης (Lloyd's, LSH, Hypercube) ***method***, μια αντιστοίχιση (map) ***idToIndexMap*** του id ενός διανύσματος με το index στο vector *points* και το ***overallSilhouette***.

PriorityQueue

Δομή μιας ουράς προτεραιότητας με αναπαράσταση vector που περιέχει Neighbor (***heap***).

CompleteBinaryTree

Δομή ενός πλήρους δυαδικού δένδρου για την αποθήκευση των καμπυλών.

Grid

Αναπαριστά ένα πλέγμα και κρατάει πληροφορία για το δ ***delta*** και το τυχαίο διάνυσμα ***t***.

Curve_

Αναπαριστά μια καμπύλη και κληρονομεί από την κλάση Data. Οι επιπλέον πληροφορίες που περιέχει είναι οι εξής: αναπαράσταση του χρόνου σε ένα vector ***tVec***, την x συντεταγμένη του πλέγματος ***gxVev***, την y συντεταγμένη του πλέγματος ***gyVev*** και το κλειδί για το hashing ***key***.

Περιγραφή προγράμματος

Τα δύο βασικά αρχεία ***search.cpp*** και ***cluster.cpp*** ακολουθούν την ίδια λογική. Κάνουν parse τις παραμέτρους της γραμμής εντολών, δημιουργούν τις κατάλληλες δομές που χρειάζονται (Hashtable, Hypercube, Cluster, Grid) αναλόγως την περίπτωση, διαβάζουν τα input αρχεία και τέλος εκτελούν τους αντίστοιχους αλγορίθμους.

Οδηγίες χρήσης προγράμματος

Μεταγλώττιση

- Όλο το πρόγραμμα: ***make***
- Πρόγραμμα search: ***make search***
- Πρόγραμμα cluster: ***make cluster***
- Unit tests: ***make tests***

Clean

- ***make clean***

Εκτέλεση

- Πρόγραμμα search: ***./build/search -i test_input_120.txt -q test_query_120 -o outputSearch.txt -k 15 -L 7 -M 1000 -probes 2 -algorithm Frechet -metric discrete -delta 2***
- Πρόγραμμα cluster: ***./build/cluster -i datasets/nasdaq2017_LQ.csv -c cluster/cluster.conf -o outputCluster.txt -update Mean -assignment Classic -complete -silhouette***
- Unit tests: ***runTests***

Github repository:

https://github.com/DImiTrisXam/algo_project1/tree/main/project2