

Ανάπτυξη Λογισμικού Για Πληροφοριακά Συστήματα

Χειμερινό Εξάμηνο 2023– 2024

Άσκηση 1

Ονοματεπώνυμο και ΑΜ:

Μαρίνα Μυλωνά 1115201900229

Βαγγέλης Βραΐλας 1115201900314

Δέσποινα Χριστοδούλου 1115201900303

Emails:

sdi1900229@di.uoa.gr

sdi1900314@di.uoa.gr

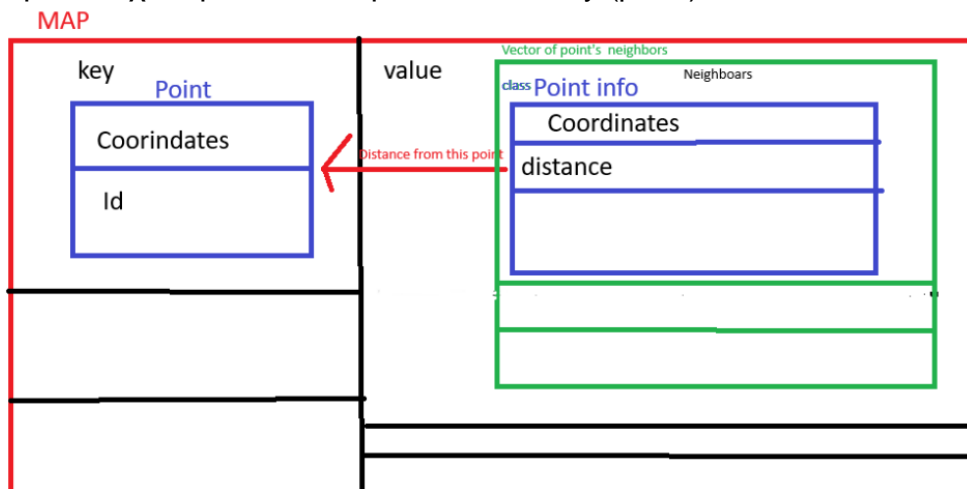
sdi1000303@di.uoa.gr

Git repository: https://github.com/Vaggelisvl/Project_2023_1

Σχεδιαστικές επιλογές:

Πιο κάτω παραθέτουμε την κύρια σχεδιαστική επιλογή, η οποία απεικονίζει το Γράφο. Συγκεκριμένα είναι ένα map, το οποίο έχει ως key την κλάση Point και ως value ένα vector από Neighbors. Οι υλοποιήσεις των custom δομών δεδομένων map και vector βρίσκονται στα αρχεία headers/structures/unorderedMap/UnorderedMap.h και headers/structures/vector/Vector.h αντίστοιχα.

Η κλάση Point αντιπροσωπεύει το κάθε στοιχείο του αρχείου που διαβάζουμε, η οποία έχει τις συντεταγμένες και το id του. Η κλάση Neighbor είναι μια προέκταση του Point με τη διαφορά ότι έχει την απόσταση του από το key (point).



Ανάγνωση από αρχείο: src/structures/Dataset.cpp

Η ανάγνωση από το αρχείο binary γίνεται στο Dataset.cpp, αφού έχει δώσει ο χρήστης το όνομα του αρχείου που θέλει να διαβάσει, τον αριθμό των διαστάσεων των σημείων και τον αριθμό των σημείων που υπάρχουν. Το τελευταίο όρισμα μπορεί να μην το δώσει, αν θέλει να διαβάσει από αρχεία με το format που μας δόθηκε γιατί θα το διαβάσει από το αρχείο.

Αρχικοποίηση Γράφου: src/structures/graph/GraphInitialization.cpp

Η μέθοδος setKRandomNeighbors εκτελεί τη διαδικασία τυχαίας αρχικοποίησης των K γειτόνων για κάθε σημείο στο Γράφο. Αφού βρει τον τυχαίο γείτονα στο σύνολο δεδομένων, υπολογίζει την απόσταση ανάμεσα στο τρέχον σημείο και τον τυχαίο γείτονα χρησιμοποιώντας τη συνάρτηση της ευκλείδεια σ απόστασης. Δημιουργεί ένα αντικείμενο Neighbors για τον τυχαίο γείτονα με το ID του, την απόσταση και τις συντεταγμένες του. Τέλος το προσθέτει στο Γράφο και κάνει το ίδιο και για τον αντίστροφο του.

Αλγόριθμος KNN: src/structures/graph/GraphInitialization.cpp

Η μέθοδος KNNAlgorithm υλοποιεί τον αλγόριθμο K-Nearest Neighbors (KNN) για τη βελτιστοποίηση του συνόλου των γειτόνων για κάθε σημείο στο γράφο. Για κάθε σημείο στο γράφο ταξινομεί τους γείτονές του με βάση την απόσταση, έτσι ώστε ο γείτονας με τη μεγαλύτερη απόσταση να βρίσκεται στο τέλος του πίνακα. Στη συνέχεια για κάθε γείτονα αναζητά τους γείτονες γειτόνων και υπολογίζει την απόσταση του μεταξύ του τρέχοντος σημείου, αφού ελέγξει για διπλά σημεία. Εάν η απόσταση αυτή είναι μικρότερη από τη μέγιστη απόσταση που έχει καταγραφεί, τότε αντικαθίσταται ο extended γείτονας. Εφόσον έχει γίνει αλλαγή, η συνάρτηση επιστρέφει 0 για να ξανακληθεί η ίδια συνάρτηση, αλλιώς επιστρέφει 1.

Αλγόριθμος KNN για σημείο εκτός Γράφου: src/structures/graph/GraphInitialization.cpp

Η μέθοδος findKNearestNeighborsForPoint, εκτελεί τον αλγόριθμο για την εύρεση των K-πλησιέστερων γειτόνων ενός query point που δεν βρίσκεται στο Γράφο. Αρχικοποιεί τυχαίους γείτονες υπολογίζοντας και τις αποστάσεις μεταξύ τους. Στη συνέχεια προσθέτει στο Γράφο το query point με τους K αυτούς γείτονες και καλεί τη πιο πάνω συνάρτηση(KNNAlgorithm). Όταν τελειώσει τον αλγόριθμο KNN αφαιρεί το γράφο το σημείο αυτό.

Δυνατότητα χρήσης εναλλακτικής μετρικής ομοιότητας: src/structure/utils/Metrics.cpp

Έχουμε υλοποιήσει δύο μετρικές, την ευκλείδεια και την manhattan οι οποίες βρίσκονται στο αρχείο Metrics.cpp .

Dataset:

1. Το format Που μας δόθηκε(input1.cpp, 00000200.bin, 00001000.bin)
2. Δικό μας format: 100 στοιχεία των 20 διαστάσεων(data.bin)

Results.txt:

Σε αυτό το αρχείο αποθηκεύονται τα αποτελέσματα από τους 20 πιο κοντινούς γείτονες κάθε σημείου του αρχείου. Χρησιμοποιείται μόνο για σκοπούς ελέγχου του αλγορίθμου KNN.

Στο library.cpp υπάρχει η main

Για compile: make all

Για τα τεστ: make test

Για να δοκιμάσουμε το library που φτιάξαμε :

export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:bin

Μεταγλώττιση του library.cpp που περιέχεται η main μαζί με την βιβλιοθήκη μας:

g++ -o main src/library.cpp -Lbin -ldataforge

Εκτέλεση για τα αρχεία με το format που μας δόθηκαν:

./main src/inpu1.bin 100

Εκτέλεση για το δικό μας αρχείο:

./main src/inpu1.bin 20 100

