



2^η Άσκηση Διαχείριση Σύνθετων Δεδομένων

Μάθημα	Διαχείριση Σύνθετων Δεδομένων (ΜΥΕ041)
Διδάσκων	Μαμουλής Νικόλαος
Ονοματεπώνυμο	Καζακίδης Θεοχάρης
ΑΜ	4679
Εξάμηνο	Εαρινό
Ακαδημαϊκό Έτος	2022-2023

Πως τρέχουμε τα προγράμματα?

→Μέρος 1

- Όνομα Αρχείου: Meros1_Kazakidis_Theocharis_4679.py
- Εντολή: `python3 Meros1_Kazakidis_Theocharis_4679.py tiger_roads.csv`

→Μέρος 2

- Όνομα Αρχείου: Meros2_Kazakidis_Theocharis_4679.py
- Εντολή: `python3 Meros2_Kazakidis_Theocharis_4679.py queries.txt`

→Μέρος 3

- Όνομα Αρχείου: Meros3_Kazakidis_Theocharis_4679.py
- Εντολή: `python3 Meros3_Kazakidis_Theocharis_4679.py queries.txt`

Τι τυπώνουν?

➔Μέρος 1

```
chareskazakides@192 Desktop % python3 Meros1_Kazakidis-Theocharis_4679.py tiger_roads.csv
0 0 [-87.752641, -87.5339075] [31.963678, 32.119055]
0 1 [-87.752641, -87.5339075] [32.119055, 32.274432000000004]
0 2 [-87.752641, -87.5339075] [32.274432000000004, 32.429809]
0 3 [-87.752641, -87.5339075] [32.429809, 32.585186]
0 4 [-87.752641, -87.5339075] [32.585186, 32.740563]
0 5 [-87.752641, -87.5339075] [32.740563, 32.89594]
0 6 [-87.752641, -87.5339075] [32.89594, 33.051317000000004]
0 7 [-87.752641, -87.5339075] [33.051317000000004, 33.206694]
0 8 [-87.752641, -87.5339075] [33.206694, 33.362071]
0 9 [-87.752641, -87.5339075] [33.362071, 33.517448]
1 0 [-87.5339075, -87.315174] [31.963678, 32.119055]
1 1 [-87.5339075, -87.315174] [32.119055, 32.274432000000004]
1 2 [-87.5339075, -87.315174] [32.274432000000004, 32.429809]
1 3 [-87.5339075, -87.315174] [32.429809, 32.585186]
1 4 [-87.5339075, -87.315174] [32.585186, 32.740563]
1 5 [-87.5339075, -87.315174] [32.740563, 32.89594]
1 6 [-87.5339075, -87.315174] [32.89594, 33.051317000000004]
1 7 [-87.5339075, -87.315174] [33.051317000000004, 33.206694]
1 8 [-87.5339075, -87.315174] [33.206694, 33.362071]
1 9 [-87.5339075, -87.315174] [33.362071, 33.517448]
2 0 [-87.315174, -87.0964405] [31.963678, 32.119055]
2 1 [-87.315174, -87.0964405] [32.119055, 32.274432000000004]
2 2 [-87.315174, -87.0964405] [32.274432000000004, 32.429809]
2 3 [-87.315174, -87.0964405] [32.429809, 32.585186]
2 4 [-87.315174, -87.0964405] [32.585186, 32.740563]
2 5 [-87.315174, -87.0964405] [32.740563, 32.89594]
2 6 [-87.315174, -87.0964405] [32.89594, 33.051317000000004]
2 7 [-87.315174, -87.0964405] [33.051317000000004, 33.206694]
2 8 [-87.315174, -87.0964405] [33.206694, 33.362071]
2 9 [-87.315174, -87.0964405] [33.362071, 33.517448]
3 0 [-87.0964405, -86.877707] [31.963678, 32.119055]
3 1 [-87.0964405, -86.877707] [32.119055, 32.274432000000004]
3 2 [-87.0964405, -86.877707] [32.274432000000004, 32.429809]
3 3 [-87.0964405, -86.877707] [32.429809, 32.585186]
3 4 [-87.0964405, -86.877707] [32.585186, 32.740563]
3 5 [-87.0964405, -86.877707] [32.740563, 32.89594]
3 6 [-87.0964405, -86.877707] [32.89594, 33.051317000000004]
3 7 [-87.0964405, -86.877707] [33.051317000000004, 33.206694]
3 8 [-87.0964405, -86.877707] [33.206694, 33.362071]
3 9 [-87.0964405, -86.877707] [33.362071, 33.517448]
4 0 [-86.877707, -86.6589735] [31.963678, 32.119055]
4 1 [-86.877707, -86.6589735] [32.119055, 32.274432000000004]
4 2 [-86.877707, -86.6589735] [32.274432000000004, 32.429809]
4 3 [-86.877707, -86.6589735] [32.429809, 32.585186]
4 4 [-86.877707, -86.6589735] [32.585186, 32.740563]
4 5 [-86.877707, -86.6589735] [32.740563, 32.89594]
4 6 [-86.877707, -86.6589735] [32.89594, 33.051317000000004]
4 7 [-86.877707, -86.6589735] [33.051317000000004, 33.206694]
4 8 [-86.877707, -86.6589735] [33.206694, 33.362071]
4 9 [-86.877707, -86.6589735] [33.362071, 33.517448]
5 0 [-86.6589735, -86.44024] [31.963678, 32.119055]
5 1 [-86.6589735, -86.44024] [32.119055, 32.274432000000004]
5 2 [-86.6589735, -86.44024] [32.274432000000004, 32.429809]
5 3 [-86.6589735, -86.44024] [32.429809, 32.585186]
5 4 [-86.6589735, -86.44024] [32.585186, 32.740563]
5 5 [-86.6589735, -86.44024] [32.740563, 32.89594]
5 6 [-86.6589735, -86.44024] [32.89594, 33.051317000000004]
5 7 [-86.6589735, -86.44024] [33.051317000000004, 33.206694]
5 8 [-86.6589735, -86.44024] [33.206694, 33.362071]
5 9 [-86.6589735, -86.44024] [33.362071, 33.517448]
6 0 [-86.44024, -86.2215065] [31.963678, 32.119055]
6 1 [-86.44024, -86.2215065] [32.119055, 32.274432000000004]
6 2 [-86.44024, -86.2215065] [32.274432000000004, 32.429809]
6 3 [-86.44024, -86.2215065] [32.429809, 32.585186]
6 4 [-86.44024, -86.2215065] [32.585186, 32.740563]
6 5 [-86.44024, -86.2215065] [32.740563, 32.89594]
6 6 [-86.44024, -86.2215065] [32.89594, 33.051317000000004]
6 7 [-86.44024, -86.2215065] [33.051317000000004, 33.206694]
6 8 [-86.44024, -86.2215065] [33.206694, 33.362071]
6 9 [-86.44024, -86.2215065] [33.362071, 33.517448]
7 0 [-86.2215065, -86.002773] [31.963678, 32.119055]
7 1 [-86.2215065, -86.002773] [32.119055, 32.274432000000004]
7 2 [-86.2215065, -86.002773] [32.274432000000004, 32.429809]
7 3 [-86.2215065, -86.002773] [32.429809, 32.585186]
7 4 [-86.2215065, -86.002773] [32.585186, 32.740563]
7 5 [-86.2215065, -86.002773] [32.740563, 32.89594]
7 6 [-86.2215065, -86.002773] [32.89594, 33.051317000000004]
7 7 [-86.2215065, -86.002773] [33.051317000000004, 33.206694]
7 8 [-86.2215065, -86.002773] [33.206694, 33.362071]
7 9 [-86.2215065, -86.002773] [33.362071, 33.517448]
8 0 [-86.002773, -85.7840395] [31.963678, 32.119055]
8 1 [-86.002773, -85.7840395] [32.119055, 32.274432000000004]
8 2 [-86.002773, -85.7840395] [32.274432000000004, 32.429809]
8 3 [-86.002773, -85.7840395] [32.429809, 32.585186]
8 4 [-86.002773, -85.7840395] [32.585186, 32.740563]
8 5 [-86.002773, -85.7840395] [32.740563, 32.89594]
8 6 [-86.002773, -85.7840395] [32.89594, 33.051317000000004]
8 7 [-86.002773, -85.7840395] [33.051317000000004, 33.206694]
8 8 [-86.002773, -85.7840395] [33.206694, 33.362071]
```

```
8 9 [-86.002773, -85.7840395] [33.362071, 33.517448]
9 0 [-85.7840395, -85.565306] [31.963678, 32.119055]
9 1 [-85.7840395, -85.565306] [32.119055, 32.274432000000004]
9 2 [-85.7840395, -85.565306] [32.274432000000004, 32.429809]
9 3 [-85.7840395, -85.565306] [32.429809, 32.585186]
9 4 [-85.7840395, -85.565306] [32.585186, 32.740563]
9 5 [-85.7840395, -85.565306] [32.740563, 32.89594]
9 6 [-85.7840395, -85.565306] [32.89594, 33.051317000000004]
9 7 [-85.7840395, -85.565306] [33.051317000000004, 33.206694]
9 8 [-85.7840395, -85.565306] [33.206694, 33.362071]
9 9 [-85.7840395, -85.565306] [33.362071, 33.517448]
chareskazakides@192 Desktop %
```

→Μέρος 2

```
chareskazakides@192 Desktop % python3 Meros2_Kazakidis_Theocharis_4679.py queries.txt
Query 1 results
13151 15262 15774 16782 21379 22260 22500 22946 22947
Cells: 1
Results: 9
-----
Query 2 results
33887 34512 34862
Cells: 1
Results: 3
-----
Query 3 results
30397
Cells: 2
Results: 1
-----
Query 4 results
1108
Cells: 1
Results: 1
-----
Query 5 results
5496
Cells: 3
Results: 1
-----
chareskazakides@192 Desktop % █
```

→Μέρος 3

```
chareskazakides@192 Desktop % python3 Meros3_Kazakidis_Theocharis_4679.py queries.txt
Query 1 results
21379 22260 22500 22946 22947
Cells: 1
Results: 5
-----
Query 2 results
34512
Cells: 1
Results: 1
-----
Query 3 results
30397
Cells: 2
Results: 1
-----
Query 4 results

Cells: 1
Results: 0
-----
Query 5 results
5496
Cells: 3
Results: 1
-----
chareskazakides@192 Desktop % █
```

Κύριες συναρτήσεις επεξεργασία:

→Μέρος 1

Το πρώτο μέρος αποτελείται από τρεις συναρτήσεις.

- **readFile(filename):** Συνάρτηση που η δουλειά της είναι να διαβάσει το αρχείο που της δίνουμε σαν όρισμα και για κάθε γραμμή του αρχείου να χωρίζει τα δεδομένα σε tokens και επιστρέφει μια λίστα με όλα τα ορθογώνια των σημείων την allMbrs και ακόμη τις ελάχιστες και μέγιστες συντεταγμένες x και y όλων των σημείων.
- **gridCells(allMbrs, minX, minY, maxX, maxY):** Συνάρτηση που ως ορίσματα παίρνει την λίστα allMbrs με όλα τα mbr που διαβάσαμε από το αρχείο και τα ελάχιστα και μέγιστα σημεία του X και Y και μέσα της δημιουργείται το grid. Αυτό ορίζεται ως ένας πίνακας κελιών 10x10, και κάθε κελί αποτελείται από δυο λίστες μια του άξονα X μια άξονας Y. Η συνάρτηση καθορίζει ποια mbr τέμνονται με κάθε κελί στο grid και δημιουργεί ένα λεξικό που κρατάει κάθε κελί που τέμνεται με mbr σε μια λίστα και τέλος η συνάρτηση επιστρέφει τις συντεταγμένες x και y των κελιών του grid και του λεξικού.
- **gridFiles(grid, minX, minY, maxX, maxY):** Συνάρτηση που ως ορίσματα παίρνει το grid που δημιούργησα παραπάνω και τα ελάχιστα και μέγιστα σημεία του X και Y. Παίρνει το λεξικό που δημιουργήθηκε από το gridCells και δημιουργεί δύο αρχεία το grid.dir και το grid.grd. Το πρώτο περιέχει πληροφορίες για το grid, συμπεριλαμβανομένων των ελάχιστων και μέγιστων συντεταγμένων x και y, καθώς και τον αριθμό των mbr που ταυτίζονται με κάθε κελί του grid. Το δεύτερο περιέχει πληροφορίες σχετικά με τα ίδια τα mbr, συμπεριλαμβανομένων των αναγνωριστικών τους, των ελάχιστων και μέγιστων συντεταγμένων x και y και των συντονισμών x και Y των σημείων εντός του mbr.

→Μέρος 2

Το δεύτερο μέρος αποτελείται από τέσσερις συναρτήσεις.

- **gridCells():** Συνάρτηση που διαβάσει τα δύο αρχεία το grid.dir και το grid.grd. Αρχικοποιεί ένα κενό λεξικό που ονομάζεται (grid) για να περιέχει τα δεδομένα που θα αποθηκεύονται σε κάθε cell του grid. Στη συνέχεια διαβάσει τις υπόλοιπες γραμμές στο "grid.dir" όπου κάθε γραμμή αντιστοιχεί σε ένα cell του grid και περιέχει ένα key και mbr. Εάν ο N είναι διάφορο του μηδεν, διαβάσει τον αντίστοιχο αριθμό γραμμών από το "grid.grd", καθεμία από τις οποίες περιγράφει ένα μόνο mbr. Αυτές οι γραμμές περιέχουν ένα id, ένα mbr και μια λίστα σημείων

.Η συνάρτηση κατασκευάζει μια λίστα mbr για το τρέχον cell και την αποθηκεύει στο λεξικό του grid με το κλειδί του cell ως κλειδί του λεξικού.

- **gridFiles(grid):** Συνάρτηση που ως ορίσματα παίρνει το grid που δημιουργήσα παραπάνω. Δημιουργεί ένα αρχείο που ονομάζεται "grid.dirTest" και γράφει τις πληροφορίες για κάθε κελί του grid σε αυτό. Εάν το κλειδί υπάρχει στο λεξικό grid, η συνάρτηση γράφει μια γραμμή στο αρχείο εξόδου που υποδεικνύει τις συντεταγμένες i και j και τον αριθμό των στοιχείων. Εάν το κλειδί δεν υπάρχει στο λεξικό grid η συνάρτηση γράφει μια γραμμή στο αρχείο εξόδου που υποδεικνύει τις συντεταγμένες i και j και το 0 ως τον αριθμό των στοιχείων .
- **readQueries:** Συνάρτηση που ανοίγει αρχείο για διάβασμα συγκεκριμένα από εκφώνηση το αρχείο queries.txt που περιέχει τα ερωτήματα παραθύρων και μετά αφού διαβάζει όλες τις γραμμές χωρίζει μέσω split (κόμμα) σε λίστα tokens όπου η θέση 1 της λίστας είναι το id και έπειτα μέσω μιας νέας λίστας tokens που τώρα χωρίζεται με space περνά τα min,max των παραθύρων και τέλος τις κάθε επανάληψης προστίθεται στην λίστα queries.
- **findResults(query, gridX, gridY, grid):** Συνάρτηση που ως ορίσματα παίρνει το grid, gridX, gridY από την gridCells() και το query. Η συνάρτηση στη συνέχεια τρέχει μέσα από τα κελιά στο grid και ελέγχει αν τέμνονται με τα παράθυρα που ορίζονται από την query. Εάν ένα κελί τέμνεται με το παράθυρο, η συνάρτηση ελέγχει κάθε mbr στο κελί για να δει αν διασταύρωσε επίσης την περιοχή και εάν ένα ισχύει τότε, το id του προστίθεται σε μια λίστα mbrs. Η συνάρτηση επιστρέφει τον αριθμό των κελιών που ελέγχθηκαν και μια λίστα mbr.

→Μέρος 3

Το τρίτο μέρος αποτελείται από πέντε συναρτήσεις.

- **gridCells():** Συνάρτηση που την εξήγησα στο μέρος 2 και χρησιμοποιείται ίδια και στο μέρος 3
- **gridFiles(grid):** Συνάρτηση που την εξήγησα στο μέρος 2 και χρησιμοποιείται ίδια και στο μέρος 3
- **readQueries:** Συνάρτηση που την εξήγησα στο μέρος 2 και χρησιμοποιείται ίδια και στο μέρος 3 μέρος
- **findResults(query, gridX, gridY, grid):** Συνάρτηση που την εξήγησα στο μέρος 2 και χρησιμοποιείται ίδια και στο μέρος 3
- **check(minX, minY, maxX, maxY, mbrMinX, mbrMinY, mbrMaxX, mbrMaxY, points) :** Συνάρτηση που ως ορίσματα έχει minX, minY, maxX, maxY, mbrMinX, mbrMinY, mbrMaxX και mbrMaxY και μια λίστα points. Ελέγχει πρώτα άμα η πλευρά X του mbr περιέχεται στο παράθυρο τότε υπάρχει

τομή του παραθύρου με το `linestring`(σχήμα A του 3.1),αντίστοιχα και όλη η πλευρά γ (σχήμα B του 3.1) ενώ σε οποιαδήποτε άλλη περίπτωση πρέπει να υπολογιστή από τον τύπο του άρθρου το αν υπάρχει τομή η όχι επομένως παίρνουμε τα ευθύγραμμα τμήματα που σχηματίζονται από τα σημεία και εφαρμόζουμε το τύπο και ελέγχουμε εάν υπάρχει τομή με κάποια ακμή του παραθύρου και αν $0 \leq t \leq 1$ and $0 \leq u \leq 1$ τότε έχουμε τομή

Παρατηρήσεις:

➔Πρόσθεσα όσα παραπάνω ουσιαστικά σχόλια μπορούσα και στα τρία μέρη του κώδικα για να είναι πιο κατανοητός τόσο ο κώδικας όσο και ο τρόπος σκέψης μου. Άμα υπάρχει πρόβλημα κατά την εκτέλεση των προγραμμάτων παρακαλώ επικοινωνήστε μαζί μου. Σας παραθέτω και να αρχείο `txt` με τις εντολές εκτέλεσης μην τυχόν και δημιουργούνται επιπλέον κενά κατά το `copy-paste` των εντολών από την αναφορά μου για την εκτέλεση που θα κάνετε