

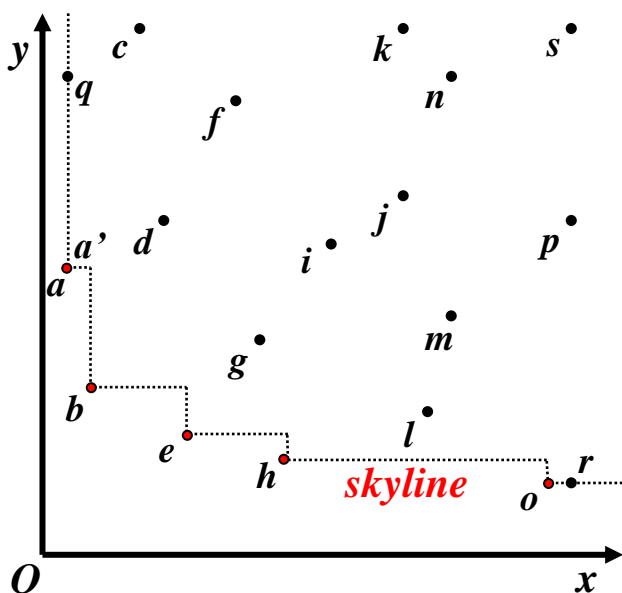
## ΑΛΓΟΡΙΘΜΟΙ 2017-2018 – ΑΣΚΗΣΗ Α (1,5 βαθμός):

### Ο αλγόριθμος Divide & Conquer για ερωτήματα κορυφογραμμής (Skyline queries)

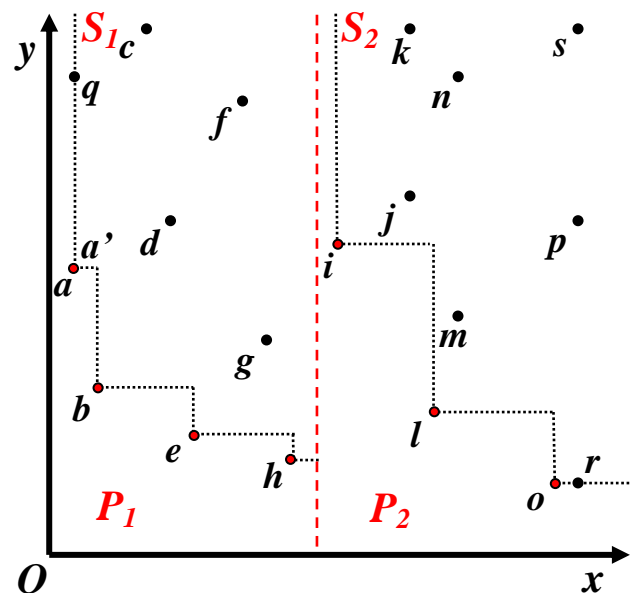
**Ορισμός:** Όταν δίνεται μία σχέση κυριαρχίας σε ένα σύνολο δεδομένων, τότε το ερώτημα κορυφογραμμής επιστρέφει όλα τα αντικείμενα τα οποία δεν κυριαρχούνται από κανένα άλλο αντικείμενο του συνόλου. Σε περίπτωση που το σύνολο δεδομένων αποτελείται από πολυδιάστατα αντικείμενα, ένα αντικείμενο θα κυριαρχεί πάνω σε ένα άλλο αν είναι το ίδιο ή καλύτερο σε όλες τις διαστάσεις και τουλάχιστον σε μία αυστηρά καλύτερο.

Τυπικά, για δύο δισδιάστατα αντικείμενα  $A(x_1, y_1)$  και  $B(x_2, y_2)$ , και θεωρώντας ότι καλύτερη τιμή είναι η ελάχιστη σε κάθε μία από τις δύο διαστάσεις, θα έχουμε ότι το  $A$  **κυριαρχεί** πάνω στο  $B$  αν και μόνον αν:  $x_1 \leq x_2$  και  $y_1 \leq y_2$  και  $(x_1 \neq x_2 \text{ ή } y_1 \neq y_2)$ .

**Παράδειγμα:** Ένα τυπικό παράδειγμα είναι όταν τα αντικείμενα είναι σημεία στο Ευκλείδειο επίπεδο, και θεωρώντας ότι η καλύτερη τιμή σε κάθε μία από τις δύο διαστάσεις είναι η ελάχιστη. Έστω τα 20 σημεία του σχήματος 1:  $a(1,12)$ ,  $a'(1,12)$ ,  $b(2,7)$ ,  $c(4,22)$ ,  $d(5,14)$ ,  $e(6,5)$ ,  $f(8,19)$ ,  $g(9,9)$ ,  $h(10,4)$ ,  $i(12,13)$ ,  $j(15,15)$ ,  $k(15,22)$ ,  $l(16,6)$ ,  $m(17,10)$ ,  $n(17,20)$ ,  $o(21,3)$ ,  $p(22,14)$ ,  $q(1,20)$ ,  $r(22,3)$ ,  $s(22,22)$ . Τότε το ερώτημα κορυφογραμμής επιστρέφει το σύνολο:  $\{a, b, e, h, o\}$ .



Σχήμα 1



Σχήμα 2

**Αλγόριθμος Ωμής-Βίας:** Ένας απλός αλγόριθμος ωμής-βίας για τον υπολογισμό της κορυφογραμμής είναι να συγκρίνουμε κάθε αντικείμενο με κάθε άλλο αντικείμενο του συνόλου δεδομένων για να δούμε αν κυριαρχείται ή όχι, χρησιμοποιώντας ένα διπλό βρόγχο (nested-loops). Ωστόσο η τετραγωνική πολυπλοκότητα του αλγορίθμου αυτού  $O(N^2)$  κάνει τον αλγόριθμο αναποτελεσματικό ( $N$  είναι το πλήθος των αντικειμένων στο σύνολο δεδομένων). Μπορεί όμως να χρησιμοποιηθεί για έλεγχο της εγκυρότητας των αποτελεσμάτων που δίνουν άλλοι αλγόριθμοι.

**Αλγόριθμος Divide and Conquer (DC):** Ένας απλός αλγόριθμος divide-and-conquer για τα ερωτήματα κορυφογραμμής είναι ο εξής: Χωρίζουμε στην μέση (ή περίπου στη μέση) το σύνολο των αντικειμένων με βάση τις τιμές τους ως προς τη μία διάσταση. Προκύπτουν δύο υποσύνολα αντικειμένων  $P_1$  και  $P_2$ . Εφαρμόζουμε αναδρομικά τον υπολογισμό των κορυφογραμμών  $S_1$  και  $S_2$  για τα  $P_1$  και  $P_2$  αντίστοιχα. Έτσι τα  $P_1$  και  $P_2$  θα χωριστούν ξανά κ.ο.κ. Ο διαχωρισμός σταματάει όταν ένα υποσύνολο περιλαμβάνει 0, 1 ή 2 σημεία. Στις περιπτώσεις αυτές ο υπολογισμός της κορυφογραμμής είναι τετριμμένος. Υπολογίζουμε την συνολική κορυφογραμμή συγχωνεύοντας τα  $S_1$  και  $S_2$ . Τότε διαγράφουμε τα αντικείμενα του  $S_2$  τα οποία κυριαρχούνται από κάποιο αντικείμενο του  $S_1$  (παρατηρούμε ότι κανένα αντικείμενο του  $S_1$  δεν μπορεί να κυριαρχείται από κάποιο του  $S_2$  καθώς ένα αντικείμενο του  $S_1$  είναι καλύτερο από κάθε αντικείμενο του  $S_2$  στη διάσταση που επιλέχθηκε).

**Παράδειγμα:** Στο παραπάνω παράδειγμα και στο σχήμα 2 έγινε διαχωρισμός των 20 σημείων σε δύο μέρη P1, P2 με βάση τη διάστασή τους x. Εφαρμόζοντας αναδρομικά τον αλγόριθμο στα δύο μέρη, οι αντίστοιχες κορυφογραμμές που επιστρέφονται είναι  $S1=\{a,b,e,h\}$  και  $S2=\{i,l,o\}$ , αντίστοιχα. Για να πάρουμε την τελική κορυφογραμμή S, πρέπει να αφαιρέσουμε τα σημεία του S2 τα οποία κυριαρχούνται από κάποιο σημείο του S1. Στο παράδειγμα αυτό τα σημεία i και l κυριαρχούνται από το h, συνεπώς θα πρέπει να διαγραφούν. Ο αλγόριθμος τερματίζει με το τελικό σύνολο  $S=\{a,b,e,h,o\}$ .

### Τι πρέπει να υλοποιηθεί

Στην εργασία αυτή πρέπει να υλοποιήσετε την αναδρομική εκδοχή του αλγορίθμου Divide and Conquer (DC) για την εύρεση της κορυφογραμμής (skyline) σε σύνολα σημείων δύο διαστάσεων στο Ευκλείδειο επίπεδο.

**Απαραίτητη προϋπόθεση για να βαθμολογηθεί η εργασία με βαθμό >0 είναι να παράγει σωστά αποτελέσματα, να μην χρησιμοποιεί ωμή βία, να υπολογίζει την κορυφογραμμή σε χρόνο το πολύ μέχρι 3 δευτερόλεπτα (για τα αρχεία εισόδου που θα δίνονται) και να σταλούν τα παραδοτέα στη σωστή μορφή.**

### Τρόπος υλοποίησης:

- 1) Να χρησιμοποιήσετε **Java 1.8 αποκλειστικά**. Όλη η άσκηση θα είναι σε ένα αρχείο java.
- 2) Το **μοναδικό** όρισμα θα είναι ένα αρχείο εισόδου με τα δεδομένα των σημείων (αρχείο κειμένου txt όπου στην πρώτη γραμμή αναγράφεται το πλήθος N των σημείων και σε κάθε επόμενη γραμμή θα είναι οι συντεταγμένες των σημείων οι οποίες είναι ακέραιοι στο διάστημα [1...1000] χωρισμένοι μεταξύ τους με κενό ή tab). Τα αρχεία εισόδου βρίσκονται στην σελίδα του μαθήματος (input10,input100,input1000,input100000,input500000).
- 3) Το πρόγραμμα να εκτυπώνει στην οθόνη τις συντεταγμένες των σημείων που ανήκουν στην κορυφογραμμή, π.χ. για το αρχείο input10.txt πρέπει να εκτυπώνει:

```
1 4
3 3
4 2
```

- 4) Ο πηγαίος κώδικας πρέπει να έχει **συνοπτικά σχόλια** μέσα στον κώδικα (inline) και **εκτενή σχόλια** Javadoc επάνω από κάθε συνάρτηση, τα οποία να εξηγούν το σκεπτικό της υλοποίησής σας και ανάλυση της πολυπλοκότητας. Τα σχόλια πρέπει να είναι ΜΟΝΟ στα αγγλικά και **όχι greeklish**.

### Παραδοτέο:

- Ένας συμπίεσμένος φάκελος που θα περιέχει αποκλειστικά **α)** το αρχείο java του πηγαίου κώδικα, και **β)** το εκτελέσιμο αρχείο jar. Ο φάκελος θα έχει το όνομα AEM.zip και θα αποσταλεί ηλεκτρονικά από το ακαδημαϊκό σας email στο βοηθό διδασκαλίας Τιάκα Ελευθέριο ([tiakas@csd.auth.gr](mailto:tiakas@csd.auth.gr)). Το email να έχει θέμα: «ALGS\_1718» και στο κυρίως μήνυμα να αναγράφετε AEM και Ονοματεπώνυμο.
- Απορίες μπορείτε να στέλνετε μέχρι τις 30/4 επώνυμα, ανώνυμα μηνύματα θα αγνοούνται.
- Επιτρέπεται η χρήση έτοιμου κώδικα ελεύθερα διαθέσιμου (έτοιμες υλοποιήσεις των αλγορίθμων με κατάλληλη προσαρμογή από εσάς) με την προϋπόθεση ότι θα συμπεριλάβετε σε σχόλια την πηγή του.
- Προθεσμία υποβολής: **Παρασκευή 4/5/2018 11:59μμ**. Θα γίνονται δεκτές ασκήσεις μέχρι και τις 6/5/2018 11:59μμ, αλλά αυτές θα έχουν μείωση βαθμού κατά 25%.

### Διευκρινήσεις:

- Εκτός από τα αρχεία που αναφέρονται ως παραδοτέα, ο φάκελος δεν πρέπει να περιέχει κάποιο άλλο αρχείο. Φοιτητές που στέλνουν netbeans, eclipse κλπ. projects ή δεν χρησιμοποιούν τον ακαδημαϊκό τους λογαριασμό θα **μηδενίζονται απευθείας**.
- Όπως αναφέρθηκε παραπάνω, επιτρέπεται η χρησιμοποίηση έτοιμου κώδικα από τρίτους με την προϋπόθεση ότι θα αναφέρεται σαφώς στα σχόλια. Όμως, όπως αναγράφεται στην ιστοσελίδα του μαθήματος, τα προγράμματα θα ελέγχονται από πρόγραμμα εντοπισμού αντιγραφών. Αν εντοπιστούν αντιγραφές μεταξύ φοιτητών, τότε οι φοιτητές θα **μηδενίζονται συνολικά στο μάθημα**.
- Στο επάνω μέρος του αρχείου με τον πηγαίο κώδικα να αναφέρονται το AEM, το όνομα και το email σας.
- Οι φοιτητές θα πρέπει να είναι έτοιμοι να δώσουν προφορικές εξηγήσεις για την υλοποίησή τους, όποτε τους ζητηθεί.
- Οι βαθμοί των ασκήσεων θα ισχύουν και για τις εξετάσεις Σεπτεμβρίου ή επί πτυχίω μέχρι να ξαναδιδαχθεί το μάθημα.

Η πιο νωρίνη σωστή υλοποίηση σε δικό μας υπολογιστή θα πάρει bonus 1 μονάδα.

### **Βιβλιογραφικές Αναφορές:**

- S.Borzonyi, D.Kossmann, K. Stocker: "The skyline operator", ICDE, pp.421-430, 2001.
- H.T. Kung, F. Luccio, F.P. Preparata: "On finding the maxima of a set of vectors", JACM, Vol.22, No.4, pp.469-476, 1975.
- F.P. Preparata, M.I. Shamos: "Computational geometry: an introduction", Springer-Verlag, New York, Berlin, 1985.