

Δομές Δεδομένων – Εργασία 3

Κακάτσος Ιωάννης 3180059

Κουτσιούκης Λάμπρος Σταύρος 3180090

Μέρος Α

Για την μέθοδο `contains()` της κλάσης `Rectangle`, υλοποιήσαμε ένα `if` το οποίο ελέγχει εάν η συντεταγμένη `x` του αντικειμένου `Point` που παίρνει η μέθοδος είναι ανάμεσα στις `xmin` και `xmax` και αντίστοιχα για την συντεταγμένη `y` να είναι ανάμεσα στις `ymín` και `ymax` που αποτελούν τις διαστάσεις του παραλληλογράμμου.

Για την μέθοδο `intersects()` της κλάσης `Rectangle`, εξετάζουμε εάν το ανώτατο σημείο του ενός παραλληλογράμμου είναι κάτω από το κατώτατο σημείο του άλλου παραλληλογράμμου ή το κατώτατο σημείο του ενός είναι πάνω από το ανώτατο σημείο του άλλου. Στη συνέχεια, εξετάζουμε εάν το δεξιότερο σημείο του παραλληλογράμμου είναι αριστερά του αριστερότερου σημείου του άλλου παραλληλογράμμου ή το αριστερότερο σημείο του ενός είναι δεξιά από το δεξιότερο σημείο του άλλου. Εάν ισχύει οποιοδήποτε από τα παραπάνω ενδεχόμενα, τότε αυτό σημαίνει ότι τα δύο παραλληλόγραμμα δεν έχουν κανένα κοινό σημείο. Σε κάθε άλλη περίπτωση τα δύο παραλληλόγραμμα τέμνονται.

Για την μέθοδο `distanceTo()` της κλάσης `Rectangle`, αφού σιγουρευόμαστε ότι το σημείο δεν είναι μέσα στο παραλληλόγραμμα (στην περίπτωση που βρίσκεται μέσα επιστρέφουμε 0), επιστρέφουμε την απόσταση του σημείου από το παραλληλόγραμμα, η οποία υπολογίζεται με τον ακόλουθο τρόπο: Νοητά προεκτείνουμε τις 4 πλευρές του παραλληλογράμμου και διακρίνουμε όσον αφορά αυτές 3 υποπεριπτώσεις. Στην πρώτη περίπτωση, αν το σημείο βρίσκεται εντός του επιπέδου που δημιουργείται από την προέκταση των 2 καθέτων πλευρών του παραλληλογράμμου, τότε επιστρέφουμε την απόσταση του σημείου από αυτό ανάλογα με το εάν βρίσκεται πάνω από το παραλληλόγραμμα ή κάτω από αυτό. Αντίστοιχα για τα σημεία που βρίσκονται στο επίπεδο που δημιουργείται από την προέκταση των 2 οριζοντίων πλευρών. Τέλος, εάν το σημείο βρίσκεται στα εναπομείναντα επίπεδα, επιστρέφουμε τη απόσταση αυτών από την κοντινότερη γωνία του παραλληλογράμμου.

Μέρος Β

Για την μέθοδο `nearestNeighbor()` της κλάσης `TreeNode`, αρχικά, ελέγχουμε εάν το δέντρο είναι άδειο. Στην περίπτωση που είναι, τότε εμφανίζουμε κατάλληλο μήνυμα και επιστρέφουμε το σημείο που δόθηκε. Αλλιώς, αφού αρχικοποιήσουμε την μεταβλητή της κλάσης `min` με 142, (το 142 είναι κάτι παραπάνω από τη μέγιστη απόσταση που μπορεί να έχει ένα σημείο από ένα άλλο σε ένα παραλληλόγραμμο μέγιστων διαστάσεων) καλούμε την μέθοδο `nearestNeighborR()`. Η μέθοδος αυτή, ξεκινά με την αρχικοποίηση της μεταβλητής `min_point` η οποία αναπαριστά το σημείο που θέλουμε να επιστρέψουμε με το πρώτο σημείο του δέντρου, δηλαδή την ρίζα. Εάν υπάρχει ο κόμβος αυτός του δέντρου, συγκρίνουμε την τρέχουσα μικρότερη απόσταση (`min`) με την απόσταση ανάμεσα στο δοθέν σημείο και στο σημείο που βρισκόμαστε. Αν λοιπόν το σημείο έχει μικρότερη απόσταση από `min`, τότε η νέα μικρότερη απόσταση γίνεται η απόσταση του σημείου που βρισκόμαστε από το δοθέν σημείο και το αποθηκεύουμε στην μεταβλητή `min_point`. Έπειτα, βάζουμε σε μία νέα προσωρινή μεταβλητή την θέση μας στο δέντρο και πηγαίνουμε αριστερά όπου καλούμε αναδρομικά την μέθοδο `nearestNeighborR()` έτσι ώστε, να κάνει το ίδιο για κάθε αριστερό υποδέντρο που προκύπτει. Η ίδια διαδικασία γίνεται και για κάθε δεξιό υποδέντρο του κάθε κόμβου.

Στη μέθοδο `rangeSearch` καλείται η αναδρομική μέθοδος `rangeSearchR`, η οποία ορίζεται με τον ακόλουθο τρόπο. Αρχικά γίνεται έλεγχος για το αν ο κόμβος `current` που έχει δοθεί ως όρισμα είναι εξωτερικός κόμβος, στην οποία περίπτωση επιστρέφεται η λίστα `container`, χωρίς να γίνει κάποια αλλαγή πάνω της από την μέθοδο. Εάν ο κόμβος δεν είναι εξωτερικός, γίνεται, πρώτα, έλεγχος για το αν το παραλληλόγραμμο `rect` τέμνεται από το παραλληλόγραμμο που αντιστοιχεί στο αριστερό υποδέντρο του κόμβου `current`, ενώ αν δεν ισχύει αυτό, ελέγχεται εάν το `rect` τέμνεται από το παραλληλόγραμμο που αντιστοιχεί στο δεξιό υποδέντρο του `current`. Εσωτερικά, στον πρώτο από τους δύο προηγούμενους ελέγχους, γίνεται, αρχικά, έλεγχος για το αν το σημείο το οποίο αντιστοιχεί στον κόμβο `current` περιέχεται στο `rect`, και εν συνεχεία, εκτελείται η ίδια διαδικασία για τα υποδέντρα του `current`. Τέλος, γίνεται εντός της διαδικασίας αυτής η αναδρομική κλήση της `rangeSearchR` για τα υποδέντρα των

αντίστοιχων κόμβων και επιστρέφεται όταν τελειώσουν οι αναδρομές η λίστα container αλλαγμένη ή απαράλλακτη, ανάλογα με το τι αλλαγές επέφεραν οι αναδρομές.