

Algoritmi 2020/21

Domača naloga 3

Nalogo morate natipkati in rešitve oddati v pdf obliki na učilnico.

Problem 1. Konveksna ovojnica.

Naj bosta S_1 in S_2 konveksni ovojnici na ravnini. Napišite algoritem, ki izračuna:

- (A) Unijo teh dveh konveksnih ovojnic. *4 točk(e)*
- (B) Preveri, če se konveksni ovojnici sekata. (Izhod programa naj bo TRUE ali FALSE)

4 točk(e)

- (C) Implementirajte oba algoritma v jeziku po vašem izboru. Vhod v program bo v txt obliki, kjer prva vrstica vsebuje x -koordinate točk prve konveksne ovojnice, druga vrstica pa vsebuje y -koordinate te ovojnice. Tretja vrstica vsebuje x -koordinate druge ovojnice in četrta vrstica y -koordinate druge ovojnice. Koordinate so podane v nasprotni smeri urinega kazalca.

Izhod naj bo sestavljen iz treh vrstic. Prva vrstica naj vsebuje x -koordinate unije konveksnih ovojnic, druga y -koordinate unije konveksnih ovojnic. Tretja vrstica naj bo TRUE ali FALSE, ki je odgovor na vprašanje: 'Ali se ovojnici sekata?'.

The output should consist of three lines. The first line shall contain x -coordinates of the convex hull points of the union, and the second line their respective y -coordinates. The third line shall be either TRUE or FALSE reflecting whether hulls intersect or no.

Spodaj je primer vhoda in pripadajočega izhoda. Oddajte tudi delujočo kodo, ki bo pokazala, da vaš program deluje pravilno. *4 točk(e)*

Vhod

0 3 2 0
 0 0 1.5 1.5
 2.5 4 5 4
 0 -1 0 1

Izhod

0 4 5 4 2 0
 0 -1 0 1 1.5 1.5
 TRUE

Problem 2. Delaunay triangulacija.

Algoritem za iskanje Delaunay triangulacije opisan v [1] je naključnostni s pričakovanim časom izvajanja $O(n \log n)$. Pokažite, da je najslabši čas izvajanja $O(n^2)$. 5 točk(e)

Problem 3. Iskanje točk z kd-drevesi (kd-tree), četrtnskimi drevesi (quadtree) in intervalnimi-drevesi (range-tree).

(A) kd-drevesa, četrtnska drevesa, in intervalna-drevesa so podatkovne strukture za hranjenje množice točk v ravnini. Vsaka izmed teh struktur ima dotično prednost pred ostalimi pri določenih pogojih. Za vsako drevo naštejte po en pogoj pri katerem ima prednost pred ostalimi dvema. Odgovore utemeljite.

4 točk(e)

(B) Naj bo P množica točk v 3-dimenzionalnem prostoru. Opišite algoritem za izdelavo osmiškega drevesa točk iz P .¹

4 točk(e)

References

- [1] Mark De Berg, Marc Van Kreveld, Mark Overmars, and Otfried Cheong Schwarzkopf. Computational geometry. In *Computational geometry*, pages 1–17. Springer, 2000.

¹Osmiško drevo (octree) je 3-dimenzionalna različica četrtnskega drevesa (quadtree).