

Algoritmi avansați

Laborator 6 (săpt. 11 și 12)

1. (1p) *Poziția unui punct față de un poligon - algoritm liniar.*

Input. Numărul de vârfuri n ale poligonului, vârfurile poligonului: $P_1 = (x_{P_1}, y_{P_1}), P_2 = (x_{P_2}, y_{P_2}), \dots, P_n = (x_{P_n}, y_{P_n})$ (în această ordine), punctul Q din \mathbb{R}^2 .

Output. Programul afișează poziția relativă a punctului Q față de poligon (în interior, în exterior, pe laturi).

Exemplu. $n = 12$, $P_1 = (0, 6)$, $P_2 = (0, 0)$, $P_3 = (6, 0)$, $P_4 = (6, 6)$, $P_5 = (2, 6)$, $P_6 = (2, 2)$, $P_7 = (4, 2)$, $P_8 = (4, 5)$, $P_9 = (5, 5)$, $P_{10} = (5, 1)$, $P_{11} = (1, 1)$, $P_{12} = (1, 6)$, $Q_1 = (3, 4)$. Pentru Q_1 afișează punctul este în interiorul poligonului. Pentru același poligon și $Q_2 = (7, 3)$ afișează punctul este în exteriorul poligonului. Pentru același poligon și $Q_3 = (3, 2)$ afișează punctul este pe una dintre laturile poligonului.

2. (1p) *Monotonia unui poligon în raport cu axele de coordonate.*

Input. Numărul de vârfuri n , vârfurile poligonului: $P_1 = (x_{P_1}, y_{P_1}), P_2 = (x_{P_2}, y_{P_2}), \dots, P_n = (x_{P_n}, y_{P_n})$ (în această ordine) din \mathbb{R}^2 .

Output. Programul stabilește dacă poligonul este x -monoton și dacă este y -monoton.

Precizare. Pentru testare, $P_1 P_2 \dots P_n$ reprezintă un poligon parcurs în sens trigonometric (acest lucru nu mai trebuie verificat). Algoritmul va avea complexitatea-timp liniară.

Exemple. (i) $n = 6$, $P_1 = (4, 5)$, $P_2 = (5, 7)$, $P_3 = (5, 9)$, $P_4 = (2, 5)$, $P_5 = (4, 2)$, $P_6 = (6, 3)$. Se afișează Poligonul nu este x -monoton. Poligonul este y -monoton.

(ii) $n = 8$, $P_1 = (8, 7)$, $P_2 = (7, 5)$, $P_3 = (4, 5)$, $P_4 = (3, 9)$, $P_5 = (0, 1)$, $P_6 = (5, 2)$, $P_7 = (3, 3)$, $P_8 = (10, 3)$. Se afișează Poligonul nu este x -monoton. Poligonul nu este y -monoton.

3. (0,5p) *Poziția unui punct față de cercul circumscris unui triunghi.*

Input. Patru puncte A, B, C, D din \mathbb{R}^2 cu A, B, C necoliniare.

Output. Programul afișează poziția relativă a punctului D față de cercul circumscris triunghiului $\triangle ABC$ (în interior, în exterior, pe cerc).

4. (0,5p) *Muchii ilegale.*

Input. Patru puncte A, B, C, D din \mathbb{R}^2 , reprezentând vârfurile unui patrulater convex.

Output. Programul indică dacă una dintre muchiile AC sau BD este ilegală.

Precizare. Pentru testare, se presupune că patrulaterul $ABCD$ este convex, sensul de parcurgere fiind cel trigonometric (acest lucru nu mai trebuie verificat).