

Zeus s-a hotărât să distrugă toți titanii de pe Pământ. El a studiat pozițiile acestora și și-a dat seama că cel mai bine ar fi să trimită un fulger care să afecteze o porțiune triunghiulară. Fulgerul va distruge toți titanii care se află în interiorul triunghiului, pe laturi sau în vârfuri.

Se cunosc numărul titanilor, coordonatele acestora, precum și coordonatele vârfurilor regiunii triunghiulare care va fi afectată de fulger.

Trebuie să se determine numărul titanilor care vor fi distruși.

Datele de intrare se vor citi din fișierul **TITANI.IN**. Acest fișier conține pe prima linie șase numere, separate prin spații, care reprezintă coordonatele regiunii triunghiulare care va fi afectată. Cea de a doua linie conține numărul **N** al titanilor, iar următoarele **N** linii conțin perechi de numere, separate printr-un spațiu, reprezentând coordonatele unui titan (toate coordonatele sunt numere reale).

Datele de ieșire se vor înscrie în fișierul **TITANI.OUT** care conține o singură linie pe care se va afla numărul titanilor distruși.

Programul trebuie să afișeze pe monitor datele de intrare (coordoanatele vârfurilor regiunii triunghiulare și coordonatele fiecărui titan).

```
Exemplu:      TITANI.IN      TITANI.OUT
              0 0 0 2 2 0      3
              4
              0 0
              0 1
              1 1
              2 2
```

Programul trebuie să afișeze datele de intrare sub următoare formă (coordoanatele punctelor sunt afișate cu câte două zecimale).

*Zona afectată este delimitată de punctele:*

*A(0.00, 0.00) B(0.00, 2.00) C(2.00, 0.00)*

*Sunt 4 titani care se află în punctele:*

*Titan 1: 0.00 – 0.00*

*Titan 2: 0.00 – 1.00*

*Titan 3: 1.00 – 1.00*

*Titan 4: 2.00 – 2.00*

*Indicație:*

Trebuie să se verifice pentru fiecare punct corespunzător unui titan, dacă punctul respectiv se află sau nu în interiorul triunghiului considerat și să se numere punctele din interiorul triunghiului.

O variantă simplă de rezolvare se bazează pe faptul că dacă un punct se află în interiorul triunghiului, pe una din laturi sau este un vârf atunci suma ariilor determinate de punctul respectiv și de perechi de vârfuri ale triunghiului dat diferă cu un  $\varepsilon$  oricât de mic (de obicei egal cu  $10^{-7}$ ) de aria triunghiului dat.

Deci pentru fiecare punct în parte vom calcula ariile triunghiurilor determinate de punctul respectiv și două vârfuri ale triunghiului. Dacă suma celor trei arii diferă cu un  $\varepsilon$  de aria triunghiului dat (adică, modulul diferenței dintre suma celor trei arii și aria triunghiului dat este mai mic decât  $\varepsilon$ ) atunci punctul se în interiorul triunghiului, pe una din laturi sau într-un vârf.

Dacă se cunosc coordonatele vârfurilor unui triunghi atunci formula de calcul pentru aria triunghiului este:

$$A = \frac{1}{2} \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} \quad (1)$$

Pentru optimizarea timpului de rulare se recomandă calcularea ariei triunghiului dat o singură dată la începutul programului.

Coordonatele fiecărui punct sunt memorate într-o structură PUNCT care conține ca membri doi reali.

Pentru vectorul care conține coordonatele titanilor se face alocare dinamică cu spațiul de memorie strict necesar și cu resetarea zonei de memorie alocate.

Funcțiile care trebuie scrise sunt descrise în baremul de notare.

**Barem de notare****Tabel nr. 1**

<b>A1</b>	Citirea coordonatelor vârfurilor triunghiului (funcția are ca parametri un tablou de structuri de tip PUNCT și un pointer la o structură FILE și nu returnează nimic)	0,4
<b>A2</b>	Citirea coordonatelor pozițiilor titanilor (funcția are ca parametri un număr întreg, un pointer la o structură FILE și returnează un pointer la o structură de tip PUNCT)	0,5
<b>A3</b>	Afișarea datelor de intrare după modelul dat în enunțul problemei (funcția are ca parametri un tablou de structuri de tip PUNCT, un pointer la o structură de tip PUNCT, un număr întreg și nu returnează nimic)	0,6
<b>A4</b>	Calcularea ariei unui triunghi (funcția are ca parametri un tablou de pointeri la structura PUNCT și returnează un real).	0,5
<b>A5</b>	Calcularea numărul de titani din interiorul triunghiului dat (funcția primește ca parametri un real, doi pointeri la o structură de tip PUNCT, un număr întreg și returnează un întreg fără semn).	0,5
<b>A6</b>	Calcularea sumei ariilor determinate de poziția unui titan și perechi de vârfuri ale triunghiului dat (funcția are ca parametri un tablou de structuri de tip PUNCT, o structură de tip PUNCT și returnează o valoare de tip real).	0,5
<b>A6a</b>	Calcularea ariei unui triunghi (funcția are ca parametri trei structuri de tip PUNCT și returnează un real).	0,5
<b>A7</b>	Deschiderea și închiderea corectă a fișierelor cu care se lucrează	0,6
<b>A8</b>	Dealocarea corectă a zonelor de memorie alocate	0,6
<b>A9</b>	Scrierea corectă a fișierului header	0,5
<b>A10</b>	Scrierea funcției main	0,6
<b>A11</b>	Funcționarea corectă a programului	1,2
<b>TOTAL TABEL 1</b>		7 p