

În fișierul *FGeom.csv* sunt informații necesare pentru memorarea unor figuri geometrice. Pe fiecare linie sunt informații despre o figură, astfel: eticheta figurii (String), lista punctelor din care este formată figura. Un punct este format din două valori, coordonată pe axa *Ox* (*double*) și coordonată pe axa *Oy* (*double*). Pentru memorarea unui punct va fi construită o clasă *Punct* cu cele două câmpuri (coordonatele pe *Ox* și *Oy*). Pentru memorarea unei figuri va fi creată clasa *Figura*, cu două câmpuri: eticheta figurii și lista de puncte (*List<Punct>*).

Să se scrie o aplicație care să îndeplinească următoarele cerințe:

1) Citirea figurilor geometrice din fișierul *FGeom.csv* într-o listă de tip *List<Figura>*. **(3 puncte)**

Structura unei linii în care sunt informațiile despre o figură este:

Eticheta,x_1,y_1,x_2,y_2,...,x_n,y_n

Criteriul de acordare a punctajului: afișarea corectă a listei de figuri la consolă.

2) Să se afișeze la consolă lista figurilor de tip patrulater (au 4 puncte). **(1 punct)**

Criteriul de acordare a punctajului: afișarea corectă a listei rezultate la consolă.

3) Să se calculeze și să se afișeze la consolă perimetrele figurilor. **(2 puncte)**

Perimetrul unei figuri se calculează ca suma distanțelor dintre două puncte consecutive, la care se adaugă distanța de la ultimul punct la primul. Distanța dintre două puncte de coordonate (x_1, y_1) și (x_2, y_2) este $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. Pentru radical se folosește metoda statică *Math.sqrt(double x)*.

Criteriul de acordare a punctajului: afișarea corectă a rezultatelor la consolă.

4) Să se sorteze lista figurilor în ordine descrescătoare după perimetru și să se salveze rezultatul sortării într-un fișier text, *Perimetre.csv*. Pentru fiecare figură se va salva eticheta figurii și perimetrul. **(2 puncte - 1 punct sortarea, 1 punct salvarea)**

Criteriul de acordare a punctajului: Vizualizarea fișierului output.

5) Să se salveze toate punctele figurilor într-un fișier binar *Puncte.dat*. **(0.5 puncte)**.

Criteriul de acordare a punctajului: Se analizează codul

6) Să se citească punctele salvate la cerința 5 într-o listă de tip *List<Punct>* și să se afișeze la consolă. **(0.5 puncte)**

Criteriul de acordare a punctajului: afișarea corectă a listei la consolă

Exemplu numeric. Pentru datele din fișierul de intrare dat, rezultatele sunt:

```
--> Cerinta 1
F1,[ (1.0,1.0) , (2.0,1.0) , (1.0,3.0) ]
F2,[ (-1.0,1.0) , (-2.0,0.0) , (-2.0,-1.0) , (-1.0,-1.0) ]
F3,[ (1.0,2.0) , (4.0,2.0) , (1.0,5.0) ]
F4,[ (1.0,-1.0) , (4.0,-1.0) , (4.0,-4.0) , (1.0,-4.0) ]

--> Cerinta 2. Patrulatere:
F2,[ (-1.0,1.0) , (-2.0,0.0) , (-2.0,-1.0) , (-1.0,-1.0) ]
F4,[ (1.0,-1.0) , (4.0,-1.0) , (4.0,-4.0) , (1.0,-4.0) ]

--> Cerinta 3. Perimetrele:
F1:5.23606797749979
F2:5.414213562373095
F3:10.242640687119284
F4:12.0

--> Cerinta 4 <Perimetre.csv>
```

F4:12.0

F3:10.242640687119284

F2:5.414213562373095

F1:5.23606797749979

--> Cerinta 6. Punctele:

(1.0,1.0)

(2.0,1.0)

(1.0,3.0)

(-1.0,1.0)

(-2.0,0.0)

(-2.0,-1.0)

(-1.0,-1.0)

(1.0,2.0)

(4.0,2.0)

(1.0,5.0)

(1.0,-1.0)

(4.0,-1.0)

(4.0,-4.0)

(1.0,-4.0)

Observatii

Nu se acordă punctaj pentru programele cu erori de sintaxă sau erori în execuție.

Punctajul se acordă dacă rezultatele sunt corecte și furnizate conform cerințelor.