7.

Az előző részben kitűzött feladatok megoldása

Feladat Készítsünk szubrutint, amely tetszőleges konvex négyszög csúcspontjainak koordinátáiból kiszámítja annak területét és kerületét. A bal körüljárás szerint következő csúcspontok koordinátáit egy mátrix egymás alatti sorai tartalmazzák. A szubrutin megfelelő értelmű szöveg nyomtatásával jelezze, ha a négy pont, vagy közülük bármelyik három egy egyenesbe esik. Ilyen esetekben a területet, kerületet ne számolja.

Megoldás. Tekintsünk a koordináta-rendszerben a csúcspontjaival megadott konvex négyszöget. Válasszunk ki minden lehetséges módon három csúcspontot úgy, hogy

– az általuk meghatározott háromszög csak egyszer forduljon elő, de

– minden ilyen háromszögben a csúcspontok bal irányú – az óramutató járásával ellentétes – körüljárási sorrendben következzenek.

Jelölje $P_n(x_n, y_n)$ a csúcspontok koordinátáit ($n = 1,\ 2,\ 3,\ 4$) és képezzük ezekkel a már korábban megismert determinánst:

$$D_l = \begin{vmatrix} x_i & y_i & 1 \\ x_j & y_j & 1 \\ x_k & y_k & 1 \end{vmatrix}$$

. Ekkor az indexre vonatkozó

l	i	j	k
1	1	2	3
2	1	2	4
3	1	3	4
4	2	3	4

táblázat biztosítja, hogy a determináns egymás alatti soraiban a csúcspont koordinátái bal körüljárás szerint következzenek.

Ha most tetszőleges négyszög csúcspontjának koordinátáit írjuk a D_l determinánsaiba akkor a következő esetek lehetségesek:

1. ha a négyszög konvex és a csúcspontokat a bal körüljárás sorrendjében adjuk meg, akkor $D_l > 0$, valamennyi l-re;

2. ha van három olyan csúcspont, amely egy egyenesbe esik, akkor van legalább egy olyan l, melyhez $D_l = 0$ tartozik;

3. ha a négyszög jobb körüljárású konvex, konkáv, vagy létezik két egymást metsző oldala, akkor legalább egy olyan l index van, amelyhez tartozó $D_l < 0$.

A programot szubrutin formájában írjuk meg, mely a hívó programból átveszi a csúcspontok koordinátáit tartalmazó tömböt. Területet, kerületet csak akkor számít, ha a négyszög konvex és bal körüljárású. Ez esetben egy jelzésül szolgáló M azonosító legyen zérus. Ha létezik négyszögünkben három egy egyenesbe eső csúcspont, akkor $\mathsf{M} = 1$ legyen, ha a 3. eset áll fenn, akkor $\mathsf{M} = -1$ értékű legyen. Szöveget nem nyomtat. A SUBROUTINE két FUNCTION-t hív, egyik a determináns értékét számítja, másik a távolságot. A program egy lehetséges formája az alábbi:

```
      SUBROUTINE QUAD(C, TER, KER, M)
      REAL KER, C(4, 2), D(4)
      DO 2 L = 1, 4
      IF(L - 2)0, 8, 9
      I = 1
      J = 2
      K = 3
      GO TO 1
8     K = 4
      GO TO 1
9     IF(L - 3)0, 0, 10
      J = 3
      GO TO 1
10    I = 2
2     D(L) = DETL(C, I, J, K)
      IF (ABS (D(L)) - 10. ** (-6)) 0, 0, 3
      M = 1
      RETURN
3     IF (D(L)) 0, 2, 2
      M = -1
      RETURN
2     CONTINUE
      TER = (D(1) + D(3)) /2.
      KER = 0.
      DO 5 I = 1, 4
      IF(I - 4)0, 6, 6
      A = C(I + 1, 1)
      B = C(I + 1, 2)
      GO TO 7
6     A = C(I - 3, 1)
      B = C(I - 3, 1)
7     KER = KER + TAAV(C(I, 1), C(I, 2), A, B)
5     CONTINUE
      M = 0
      RETURN
      END
```

Programrészünk további két szubrutint használ. Ezek közül a FUNCTION TAAV már szerepelt az 5. részben. A másik a FUNCTION DETL az alábbi:

```
      FUNCTION DETL(W, L, M, N)
      DIMENSION W(4, 2)
      A = W(L, 1) * W(M, 2) + W(L, 2) * W(N, 1) + W(M, 1) * W(N, 2)
      DETL = A - W(N, 1) * W(M, 2) - W(N, 2) * W(L, 1) - W(L, 2) * W(M, 1)
      RETURN
      END
```

4. 3. (folytatás)

Mint említettük, a SUBROUTINE-t külön e célra szolgáló utasítással kell hívni. Előző rovatunkban közölt MAGAS azonosítójú szubrutin hívására pl. a

```
      CALL MAGAS (COR, VMAG, PMAG, N)
```

utasítás szolgálhat. CALL az utasítás jele, MAGAS az ezt követő SUBROUTINE-azonosító. Ezután zárójelben következnek az aktuális paraméterek. Ezek közül példánkban csak a COR tömbnek kell értékkel rendelkeznie a szubrutin hívásának pillanatában. A példabeli szubrutin nyitó utasítása

```
      SUBROUTINE MAGAS (CS, VON, MP, N)
```

volt. A szubrutin CS tömbje átveszi a COR-ban tárolt adattömböt. Ezután értéket ad a VON, MP és N azonosítójú vektoroknak, illetve számnak. Amikor a szubrutin végrehajtása befejeződik, a vezérlés visszatér a hívóprogramra, amelyben a számított értékekkel most már rendre a VMAG, PMAG és N azonosítójú vektorok, illetve szám rendelkezik.

Példát mutatunk, melyben értékadó utasítások határozzák meg egy háromszög csúcspontjainak koordinátáit. A hat paramétert hat különféle függvénnyel **12** menetben változtatjuk. A függvényekre itt most az jellemző, hogy azok egyik változója a változtatni kívánt koordinátának az előző menetben kapott értéke, másik változója a menetet számláló azonosító aktuális értékének valós típusú alakja:

$$z_i = f(z_{i-1}, i),$$

ahol i a menetet számláló szám, z_i valamelyik x vagy y koordinátaérték ebben a menetben, z_{i-1} ugyanez az előző menetben. Az ilyen típusú összefüggés az ún. rekurziós formulákhoz tartozik, melyeknek a programozási feladatok egy részében fontos szerepük van.

Programunk először kinyomtatja a pontok koordinátáit ilyen formában:

	CSUCSP.	KOORD.
P1	−3.0000	1.0000
P2	3.0000	−2.0000
P3	4.0000	4.0000

Ezt követik sorban a magasságvonalak hosszai, a következő sorban a magasságpont két koordinátája. Mindegyik elé szöveget nyomtatunk. Azt akarjuk, hogy egy lapra csak három háromszögnek az adatai kerüljenek. E célból programunkba a DO utasítás után az

```
| | IF(L+2-((L+2)/3)*3)2,0,2
```

utasítást írtuk. Ebben L a ciklust számláló egész típusú szám. Tudni kell, hogy egész típusúak osztásának eredményéül a hányadosnak csak az egész részét kapjuk, egész típusú szám formájában. Pontosabban ha i és j egészek, akkor

$$i/j \text{ eredménye } sg\left(\frac{i}{j}\right) \cdot \left[\left|\frac{i}{j}\right|\right],$$

ahol a sg az előjelfüggvény, a szögletes zárójel az egészrész, másképpen entier függvény jele. A fentiek szerint L értékei mellett a zárójelben álló kifejezések értékei az alábbi táblázat szerint alakulnak:

L	(L+2)/3	((L+2)/3)*3	L+2−((L+2)/3)*3
1	1	3	0
2	1	3	1
3	1	3	2
4	2	6	0
5	2	6	1
6	2	6	2
7	3	9	0
8	3	9	1
9	3	9	2
10	4	12	0
11	4	12	1
12	4	12	2

Valahányszor a jobb oldali kifejezés helyettesítési értéke zérus, lapváltást végző WRITE-ra ugratunk. A program egy lehetséges előállítása az alábbi:

```
      MASTER PTIZ
      DIMENSION COR(3,2),VMAG(3),PMAG(2)
      COR(1,1) = -3.
      COR(1,2) = 1
      COR(2,1) = 3.
      COR(2,2) - 2.
      COR(3,1) = 4.
      COR(3,2) = 4.
      DO 1 L = 1,12
      IF(L + 2 - ((L + 2)/3) * 3)2,0,2
      WR1TE(3,6)
    6 FORMAT(1H1,6(/))
    2 WRITE(3,7)(I,(COR(I,J),J = 1,2),I = 1,3)
    7 FORMAT(19X,14HCSUCSP.KOORD.//3(10X,1HP,I1,6X,2F12.4/))
      CALLMAGAS (COR,VMAG,PMAG,N)
      IF(M)0,0,5
      WRITE(3,3)((VMAG(I),I = 1,3),(PMAG(J),J = 1,2))
    3 FORMAT(//19X,9HMVONALAK :,3F10.3//19X,13HMPONT KOOR - K :,
     x2F10. 3,5(/))
      GO TO 8
    5 WRITE (3,4)
    4 FORMAT(///10X,28HA PONTOK EGY EGYENESBE ESNEK,5(/))
    8 V = FLOAT(I)
      COR(1,1) = -COR(1) + V
      COR(1,2) = COR(1,2) - V
      COR(2,1) = (COR(2,1) + V)/V
      COR(2,2) = -COR(2,2)/V
      COR(3,1) = COR(3,1) - V
      COR(3,2) = (COR(3,2) + V)/(V + 1.)
    1 CONTINUE
      STOP
      END
```

4.4 A szegmensek, szegmentálás

Az olyan FORTRAN nyelvű programrészeket, melyek azonosítóikban, címkéikben egymástól függetlenek, de egy összefüggő programozási feladat részeit képezik, **programszegmensek**nek nevezzük. A programszegmenseket a fordítóprogram külön-külön fordítja le.

A szubrutinok közül a FUNCTION és a SUBROUTINE szegmensek, létezik azonban más típusú szegmens is, mellyel rovatunkban nem foglalkozunk. A szubrutinokat és szegmenseket két halmaznak tekintve, a FUNCTION és SUBROUTINE ezek metszetébe tartozik.

Minden FORTRAN programban van egyetlen olyan szegmens, ahonnan a számítás végrehajtása megindul. Ezt **főszegmens**nek szoktuk nevezni. Rovatunkban a főszegmens nyitó utasításának jelölésére a MASTER-t használjuk, de vannak olyan gépi reprezentációk, melyekben ez más (LV: 133. old.).

Programok készítésének, ellenőrzésének, javításának egyszerűbbé, gyorsabbá tételének érdekében a programokat célszerű szegmentált formában készíteni még olyan esetekben is, amikor a szegmensek nem a szubrutinok feladatait töltik be. Programok szegmentálásával, általánosabban struktúrálásának vizsgálatával, fejlesztésével napjainkban is foglalkoznak a szakemberek.

Feladat

a) SUBROUTINE-ok készítendők, melyek egy háromszög csúcspontjainak koordinátáiból kiszámítják
– a területet és kerületet,
– a súlyvonalak hosszát és a súlypont koordinátáit,
– a háromszög köré írható kör középpontjának koordinátáit és a kör sugarának hosszát.

b) Program írandó, melyben a csúcspontokat értékadó utasításokkal határozzuk meg, majd felhasználva az elkészített SUBROUTINE-okat, kinyomtatja az összes számított eredményt megfelelő feliratokkal.

A Számítástechnikai Rovat kitűzött feladatainak megoldásában sokan kitűntek.

Ezek közül 100 – 100 Ft jutalomban részesülnek a következők: *

Horváth Nándor, *Soós József*, *Nagy Péter* (Földes Ferenc Gimnázium, Miskolc), *Micskó Zoltán* (Árpád Gimnázium, Tatabánya), *Grolmusz Vince* (Apáczai Csere János Gyakorló Gimnázium, Budapest), *Bozóky-Szeszich Elek* (Berzsenyi

*A jutalmakat könyvutalvány formájában küldjük el az iskolákba.

Dániel Gimnázium, Budapest), *Nagy Tibor* (Piarista Gimnázium, Budapest), *Madi Tibor* (Katona József Gimnázium, Kecskemét), *Kovács Zsolt* (Berzsenyi Dániel Gimnázium, Sopron), *Lukács Edit* (Alpári Gyula Közgazdasági Szakközépiskola, Eger).

Dicséretben részesülnek:

Szádvári Éva, *Valiskó Hedvig* és *Kökény Gabriella* (Alpári Gyula Közgazdasági Szakközépiskola, Eger), *Rudolf Éva*, *Csesznak Mihály* és *Mihályfi Gyula* (Földes Ferenc Gimnázium, Miskolc), *Gottlieb Ágnes* és *Végh Ferenc* (Katona József Gimnázium, Kecskemét), *Maga Ferenc* (Temesvári Pelbárt Ferences Gimnázium, Esztergom), *Tábori László* (Landler Jenő Gimnázium, Nagykanizsa).

A feladatok előkészítésében nyújtott segítségért köszönetet mondok a budapesti Móricz Zsigmond Gimnázium két III. osztályos tanulójának: *Boros Tamás*nak és *Mernyei Ferenc*nek.