

### 5.3 A READ-be épített ciklus

Ezt olyankor használjuk, amikor tömböt olvastatunk be. Formailag megegyezik a WRITE-ba épített ciklussal. A READ első zárójelpárját követő READ-listát ilyenkor összetettnek mondjuk. Használatára néhány példát mutatunk be.

– Tegyük fel, hogy egy kártyára 10 adatot lyukasztottunk, egyenként 8 karakter szélességű mezőkbe. Ha ezeket egy 10 elemű, egy dimenziós tömbbe akarjuk betölteni, akkor pl. az alábbi utasításpárt használhatjuk:

```
2 | | READ(1,2)(A(I), I = 1, 10)
  | | FORMAT(10F8.2)
```

– Az előzőhöz hasonló módon lyukasztott, 7 db kártyáról most egy 70 elemű vektort töltünk fel. A FORMAT-ban előírjuk, hogy hány adat beolvasása után kell újabb kártya olvasására áttérni. A kártyaváltás jele a READ-hez tartozó FORMAT-ban a /, azaz a per jel. Az utasítópár most az alábbi lehet:

```
4 | | READ(1,4)(B(I), I = 1, 70)
  | | FORMAT(6(10F8.2/), 10F8.2)
```

Láthatjuk, hogy a per jel elhelyezése miatt csak 6 az ismétlési tényező, mivel végrehajtási hibát, esetleg a program megakadását idézi elő, ha az utolsó szabályosan beolvasott kártya után újabb kártya váltását írjuk elő, most már olyan kártyára, amely nem tartozik a szóban forgó READ-hez. Emiatt írtuk le ismételten a hetedik kártya adatainak specifikációját.

– Mátrix betöltésére szolgál a soron következő példa. 25 db kártya mindegyikén 5 db, egyenként 12 karakter széles mezőn valós típusú számok állnak. A beolvasásnál 25 soros, 5 oszlopos mátrixot töltünk be. Az erre szolgáló utasítások lehetséges alakja:

```
6 | | READ(1,6)(C(I,J), J = 1, 5, I = 1, 25)
  | | FORMAT(24(5F12.3/), 5F12.3)
```

– Egy 8 soros, 12 oszlopos mátrixot töltünk be úgy, hogy egy-egy sor adatait 3 kártyáról olvassuk be. Ezek közül az első kettőn 5–5, a harmadikon 2 adat áll. Valamennyi valós típusú szám, egyenként 16 karakter széles mezőn helyezkedik el. A megoldást az alábbi utasítások jelenthetik:

```
10 | | READ(1,10)(D(I,J), J = 1, 12, I = 1, 8)
    | | FORMAT(7(2(5F16.5/), 2F16.5)
    | S | 2(5F16.5/), 2F16.5)
```

A beépített ciklus számlálónak csak számértéket írjunk az utasításba, azonosítót ne használjunk helyettük. Példáink csak soronkénti beolvasást mutattak be, hasonló módon oszloponként is lehetséges a beolvasás.

Példa:

7 élelmiszerbolt mindegyikébe 3 termelősövetkezet (A, B és C jelűek) szállít árut. A tsz-ek és a boltok között az egy mázsa árura jutó átlagos szállítási költséget az alábbi táblázat mutatja:

	élelmiszerboltok						
	1	2	3	4	5	6	7
„A” tsz	2,53	6,36	4,49	1,81	5,07	3,70	3,45
„B” tsz	5,46	4,51	1,53	0,77	6,19	5,92	2,25
„C” tsz	3,51	7,04	2,80	3,18	1,88	2,16	5,37

ahol az átlagos szállítási költség Ft/q-ban értendő. Egy adott napon a boltok rendelése az alábbi mennyiségek szállítását írja elő (q-ban):

	élelmiszerboltok						
	1	2	3	4	5	6	7
„A” tsz	9,31	6,27	2,98	8,34	3,69	7,11	6,99
„B” tsz	15,43	10,40	10,32	11,50	19,40	13,31	9,83
„C” tsz	17,07	16,71	19,03	10,10	7,07	11,09	16,03

Program készítendő, amely a két tömböt a kártyáról beolvassa és kinyomtatja, majd kiszámítja és kinyomtatja

- az adott rendeléshez tartozó szállítási költség táblázatát,
- az egyes tsz-ekre az áruszállítások összköltségeit,
- az egyes boltokra az áruszállítások összköltségeit.

Egy lehetséges program az alábbi:

```

MASTER KOZERT
DIMENSION A(3, 7), B(3, 7), C(3, 7), X(3, 7), Y(3, 7)
1 READ(1, 1)((A(I, J), J = 1, 7, I = 1, 3)
  FORMAT(2(7F6.2/), 7F6.2)
2 READ(1, 2)((B(I, J), J = 1, 7), I = 1, 3)
  FORMAT(2(7F10.2), 7F10.2)
  DO 3 I = 1, 3
  DO 4 J = 1, 7
4 C(I, J) = A(I, J) * B(I, J)
3 CONTINUE
  DO 5 I = 1, 3
  X(I) = 0.
  DO 6 J = 1, 7
6 X(I) = X(I) + C(I, J)
5 CONTINUE
  DO 7 J = 1, 7
  Y(J) = 0.
  DO 8 I = 1, 3
8 Y(J) = Y(J) + C(I, J)
7 CONTINUE
  WRITE(3, 9)((A(I, J), J = 1, 7), I = 1, 3), (B(I, J),
P J = 1, 7), I = 1, 3), (Y(K), K = 1, 7))
O (X(K), K = 1, 3), (Y(K), K = 1, 7))
9 FORMAT(1H///3(20X, 7(F7.2, 7X)/), 2(///3(20X,
R 7(F11.2, 3X)/), ///10X, 15HAZ A, B EES C TSZ,
S 39H – EK ARUSZALLITASAINAK OESSZKOELTSEEGERI :,
T 3(F11.2, 10X)///10X, 14HAZ 1 – 7 SZAAMU,
U 40HBOLTOKBA TOERTEENT AARUSZAALLITAASOK OES,
V 14HSZKOELTSEEGERI :, //20X, 7(F11.2, 5X))
  STOP
  END
  FINISH

```

Feladatok:

1. Az alábbi feladatok legfeljebb  $10 \times 10$  méretű, egész típusú számokat tartalmazó tömbre vonatkoznak.

a) Szubrutin készítenőd, amely  $n$  soros,  $m$  oszlopos mátrixban ( $2 \leq n \leq 10$  és  $2 \leq m \leq 10$ ) fölcseréli az  $i$ -edik sor elemeit a  $j$ -edik sor elemeivel, ahol  $i \neq j$ ,  $l \leq i \leq n$  és  $1 \leq j \leq m$ . A szubrutin átveszi a mátrixot, és az  $i$  és  $j$  számokat és az átvett mátrixot módosítva adja vissza.

b) A fentihez hasonló megkötésekkel szubrutin készítenőd oszlop cseréjére.

c) Program készítenőd, amely felcseréli a legnagyobb sorösszegű sort a legkisebbel, valamint a legnagyobb oszlopösszegű oszlopot a legkisebbel. Nyomatandó az eredeti mátrix, a cserélt sorú mátrix és (az eredetihez mérten) cserélt oszlopú mátrix.

2. Beküldendő a példában bemutatott programhoz tartozó, helyesen kitöltött adatlap.

Cím: Dr. ADA-WINTER PÉTER, Munkaügyi Minisztérium Számítástechnikai Intézet, 1089 Bp. Reguly Antal u. 57-59.