

## EGY INTERAKTIV PROGRAMSZERKESZTÉSI MÓDSZER ISMERTETÉSE

Ratkó István

MTA SZTAKI

A SZERKESZ nevű fortran program /SzP/ egy ún. kiinduló programból /röviden KP/ a felhasználó által megadott szám adatok, karakter adatok alapján összeállít két /szükség esetén egy/ fortran programot, az ún. célprogramokat /CP/.

A szám adatok, karakter adatok, továbbá az, hogy a KP mely sorai kerüljenek át a CP-ba, bizonyos feltételek figyelembevételétől függhetnek. Nézzük meg pontosabban, mit jelent ez.

Először is tisztázni kell a "feltétel figyelembevételének" fogalmát. A felhasználó szabhat pl. ilyen feltételt: a készítendő listán csak olyan betegek legyenek, akik legalább egy éve /365 napja/ okésok.

Ezt egy megfelelő utasítássorozatnak a CP-be történő beépítésével érhetjük el. Természetesen, ha nem akarja a felhasználó figyelembe venni ezt a feltételt, az utasítássorozat nem szerepelhet a CP-ben.

A KP a következő strukturájú:

```
-----  
1. feltételhez tartozó programsorok  
-----  
  :  
-----  
n. feltételhez tartozó programsorok  
-----
```

Az ábrával kapcsolatban a következőket jegyezzük meg:

- (i) A KP nem minden sora kerül át a CP sorai közé.
- (ii) A ----- egy vagy több olyan programsort jelöl, amely mindig átkerül a KP-be.
- (iii) Egy adott feltételhez több helyen is tartozhatnak programsorok.
- (iv) Milyenek lehetnek a feltételekhez tartozó programsorok?
  - (a) Egyszerű utasítás, mely a feltétel figyelembevétele esetén automatikusan átkerül a CP-be.
  - (b) Olyan utasítás, amely tartalmaz a felhasználó által változtatható adatot, pl.

IF(IDO.GT.365)GOTO1

Ebben a sorban a 365 a változtatható adat. Ha az IDO változóban tároljuk azt, hogy hány napja OKÉ-s a beteg, az 1 címkejű utasítás az input file-ről történő beolvasás, akkor a fenti utasítás azt szabja meg, hogy a 365 napnál régebbi OKÉ-sok ne legyenek rajta a listán. Természetesen, ha a 365 napos okézás "soknak" bizonyul, a felhasználó azt szeretné, hogy ez a határ pl. 180-ra

legyen változtatható. Ennek biztosításához a KP-ben a fenti programsort így írjuk:

```
IF(IDO.GT.  
1XXXXXX  
2)GOTO1
```

Az aktuálisan szükséges számot az XXX-es sor helyére fogja az SzP berakni.

(c) Un. "utasításcsoport", pl.

M(1)=1

M(2)=1

⋮

M(28)=1

Az ilyen utasításcsoportnak akkor van szerepe, amikor pl. a 28 műtéti diagnózis közül csak a 7., 10., 12. és 13. műtét jöhet számításba. Azaz, a készítendő listára csak olyan betegeket akarunk rávenni, akiket a fenti diagnózisok valamelyikével műtöttek. Ekkor a fenti utasításcsoportból csak a 7., 10., 12. és 13. sorok kerülnek a CP-be.

Ugy is fogjuk ezt mondani, hogy az M(.) tömbből a 7., 10., 12. és 13. tömbelemekre van szükség.

A listára kerülés feltétele ugyanis /CP-ben/  $M(I)=1$ . Pontosabban, ha valakinél a négy diagnózis D1, D2, D3, és D4, az akkor és csak akkor lehet a listán, ha

$M(D1)=1$  vagy  $M(D2)=1$  vagy  $M(D3)=1$  vagy  $M(D4)=1$  /kevesebb diagnózisnál értelemszerűen csökken a diszjunkciók száma/.

Nézzük meg ezek után, hogy a KP milyen módon ad információt a SzP számára arra vonatkozólag, hogy mi és hogyan kerüljön a CP-be.

Ezeket az információkat a KP 73.-75. pozícióiban helyezzük el az alábbi módon:

Ha azt akarjuk, hogy a programsor minden feltétel nélkül bekerüljön a CP-be, a 73.-75. pozíciókat üresen hagyjuk. Egyéb esetben a 73.-74. pozíció a CP kialakításában résztvevő feltétel sorszáma /max. 50 feltétel lehet/. A 76. pozíción lévő szám értéke és jelentése a következő: a, 1; a programsor változtatás nélkül kell b, 2; utasításcsoport 1. utasítását jelöli c, 3; utasításcsoport nem első utasításait jelöli d, 4; ehhez a programsorhoz tartozik változtatható számadat.

A SzP a következőképpen működik:

1. Felteszi a kérdéseit a felhasználónak:

..... ?

A felhasználó válasza:  $i$  vagy  $n$  vagy  $v$ , melyeknek jelentése:

$i$ : figyelembe kell venni a ? előtti feltételt a CP összeállításánál

$n$ : nem kell figyelembe venni

$v$ : nem kell figyelembe venni, sőt a további feltételeket sem.

Bármilyen más ■ válasz esetén a gép kiírja:

■ volt a válasza, s ez rossz  
ujra felteszem a kérdést

..... ?

2. Miután a felhasználó már minden kérdésre válaszolt vagy valamelyikre *v* választ adott, a gép megkérdezi:

tehát ön a következőkre válaszolt igennel:

....?  
....?  
....?

Vagyis felsorolja az összes olyan feltételt, amelyet a felhasználó figyelembe akar venni a CP-ben.

Erre a felhasználó válasza:

i: igen, ekkor a program futása folytatódik;

■ /bármilyen más/ a program leáll, a CP összeállítása nem történik meg.

3. Ezek után elkezdődik a CP összeállítása. A SzP egyenként, egymás után megvizsgálja a KP sorait. Ha a 73.-75. pozíció üres, akkor ezt a sort átmásolja a CP-be. Ha nem üres, megnézi mi áll a 73.-74. pozíción. Amennyiben ott olyan feltételnek a sorszáma áll, amit nem kell figyelembe venniünk /a megfelelő kérdésre *n* vagy *v* választ adtunk/, akkor ezt a sort nem másolja át a CP-be. Ha a feltételt figyelembe kell vennünk, akkor a továbbiak a 75. pozíción álló számtól függően más-más módon alakulnak.

a. A pozíción 1 áll. A sort az SzP átmásolja a CP-be.

b. A pozíción 2 áll. A gép kérdése:

Kérem a .....? feltételhez tartozó tömbelemeket.

Erre 40I2 formátumnak megfelelően az alábbiakat kell válaszolni:

hány tömbelemre van szükség  
melyek ezek a tömbelemek

A 36. oldalon említett példában a válasz:

4 7101213

A SzP ettől a sortól kezdve a 2 és 3 /75. pozíció!/ jelű sorokból a megfelelő tömbelemekhez tartozó sorokat átmásolja a CP-be.

c. A pozíción 3 áll. Erről az előző pontban beszéltünk.

d. A pozíción 4 áll. A gép kérdése:

Kérem a ...? feltételhez tartozó számot.

Erre I6 formátumnak megfelelően kell válaszolni. A beütött számmal a KP

1XXXXXX

sorában az XXXXXX karaktersorozatot helyettesíti, s az így kapott új sort viszi a CP-be.