

Demográfiai modellek

0. A modell rövid ismertetése

A modellünkkel megvizsgáljuk, hogy hogyan függ egy populáció **egyedszáma** és **korcsoport-eloszlása** a korcsoportokhoz rendelt numerikus jellemzőktől. A jellemzők a következők: a „**szülési hajlandóság**” (1 utód születésének valószínűsége, vagy utódok várható száma, mindezt időegységre vonatkoztatva), „**halandóság**”. Az egyes korcsoportokba az azonos korúakat soroljuk, a kort a modell választott időegységében mérjük.

Az **időegység** a populáció életében a ciklushossz. Egy időegység alatt minden egyeddel 3 féle esemény történhet: **meghalhat**, **szülhet**, egy időegységnyit **öregedhet**. E mellett a populáció egészére „globális” hatások is vonatkozhatnak. Pl. járványok, mesterséges beavatkozások. Az **időegységet továbbbonthatatlannak** vesszük, ami azt jelenti, hogy nincs értelme azt fontolgatni, hogy az időegység elején vagy a végén történik az egyeddel annak esetleges halála, hanem el kell dönteni, hogy a halálozás eseményével, vagy a szülés eseményével kezdjük a szimulációt. Az első választása esetén már szülésre nem kerülhet sor az adott korhoz tartozó időegységben, vagyis az utolsó korcsoportot garantáltan szülésképtelenségre kárhoztatjuk. A másik esetben kizárjuk annak lehetőségét, hogy a szülés előtt meghalhasson az egyed, vagyis a legfiatalabbak halandósága ellenére a szülésképesége megmarad, ami statisztikai torzulást hoz be a modellbe. Jobban járunk, ha a halálozással kezdjük a szimulációt.

A tapasztalatokhoz igazodva modellünkkel vizsgálni fogjuk azt az esetet is, amelyben a **korcsoportjellemzők** (szülési és halálozási ráták) **függenek az összegyedszámtól**. Vegyük majd észre, hogy ezen adatok más paramétereiktől is függővé tehetők, pl. –szintén a valós helyzetnek megfelelően– az időtől.

1. Elvárások

A modellnek nagyszámú paramétere lehet, hiszen csak korcsoportonként kettőre biztosan szükség lesz. Ezért hasznos eleve úgy tervezni a programunkat, hogy legyen képes akár **klaviatúráról**, akár **fájlból** beolvasni az adatokat.

Mivel az **utolsó korcsoportból** továbblépés nem lehetséges, ezért a hozzá tartozó **halálozási ráta 1** kell, legyen, s mivel így már szülni sem tud, ezért a **szülési ráta 0**. Ezeket tehát **nem kell beolvasni!**

A programunkkal szembeni használati elvárásainkat legjobban az alábbi, futás során keletkezett ábrásor foglalja össze.

```

Demográfia -- paraméterek beolvasása
Fájlnev <ha "", akkor konzol>:
Induló létszám:100
Korcsoport-szám:5
Halálozási ráták:
.1
.05
.1
.15
Szülési ráták:
.1
.3
.5
.3
    
```

1. ábra. Egy futási kép – „Kézi” paraméterezés, a fájlnev: üres vagy „CON”.

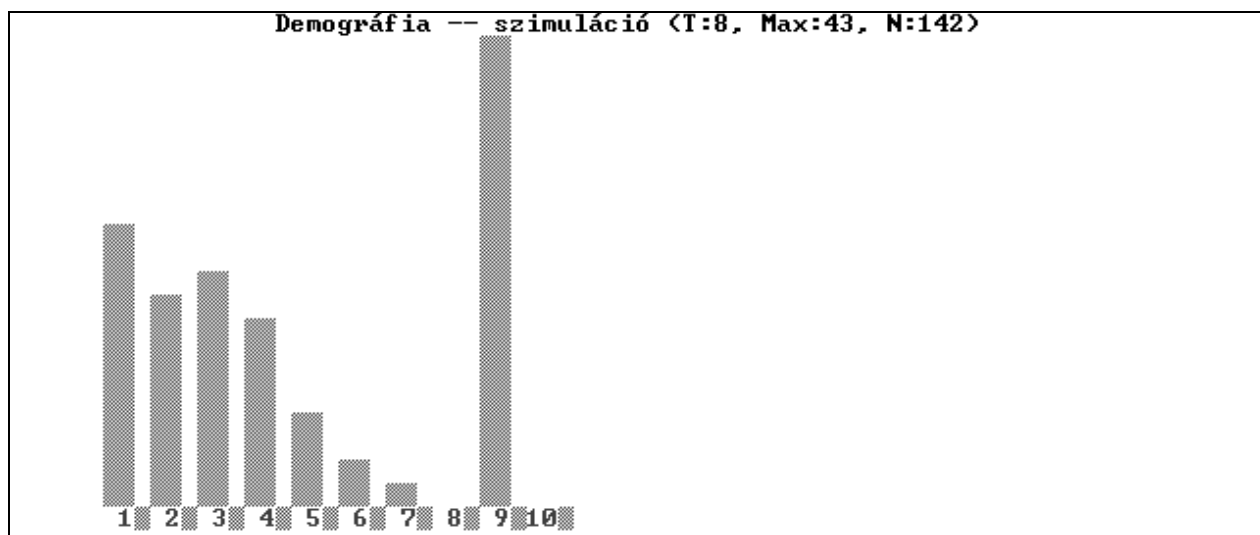
Fájlból történő indításkor a beolvasott paramétereket –ellenőrzési célból– kilistázzuk.

```

Demográfia -- paraméterek beolvasása
Fájlnev <ha "", akkor konzol>:demogl.par
Induló létszám:100
Korcsoport-szám:10
Halálozási ráták:
 0.2000 0.1000 0.1000 0.0500 0.0500 0.1000 0.1500 0.2500 0.4000
Szülési ráták:
 0.0000 0.1000 0.1500 0.2500 0.4000 0.5000 0.4000 0.3000 0.1500
    
```

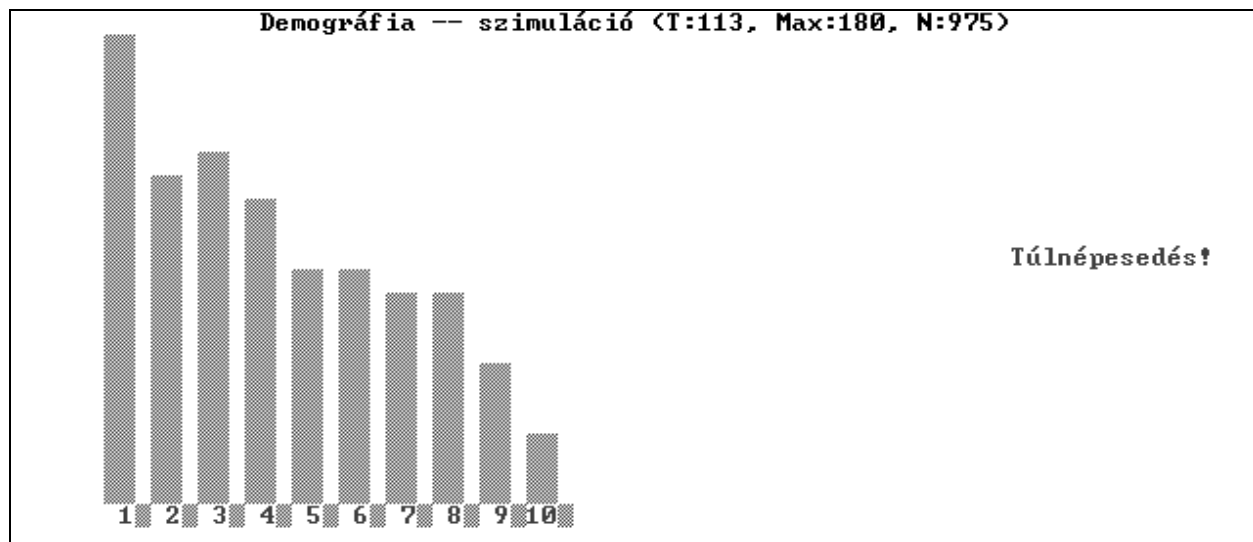
2. ábra. Egy futási kép – Indítás egy paraméterfájl megadásával (demogl.par).

A program érdemi működése során **kiírja** a modell legfontosabb **állapothatározóit**: a populáció méretét (N) és a legnépesebb korcsoport egyedszámát. A teljesebb kép kedvéért kirajzolja a korcsoport-eloszlást is.



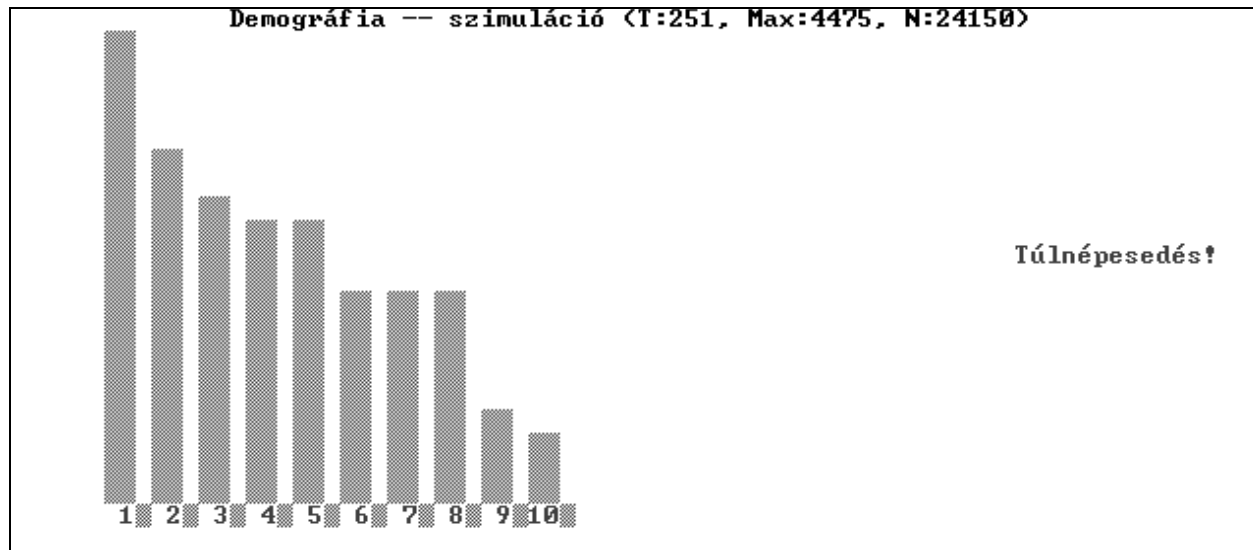
3. ábra. Egy futási kép – Valamikor a kezdetén, távol az egyensúlytól.

A modell működése során beleütközhet memóriakorlátokba. Ez akkor következik be, amikor az egyedeket tároló tömb mérete túlnő a deklarációkor megadott maximumon. Ez a „**túlnépesedés**” jelensége.



4. ábra. Egy futási kép – Egy „túlnépesedési” szituáció. (MaxN=1000 belső paraméter esetén.)¹

Ekkor azonban a futás folytatódhat; nyilván: némi csalás árán: korcsoport-arányosan csökkentve a populáció méretét, miközben az tényleges (csökkentés nélküli) összlétszámot adminisztrálva és kijelezve. Így a minket érdeklő minkét kérdésre (korcsoport-eloszlás és összegyedszám) a választ tudunk adni.



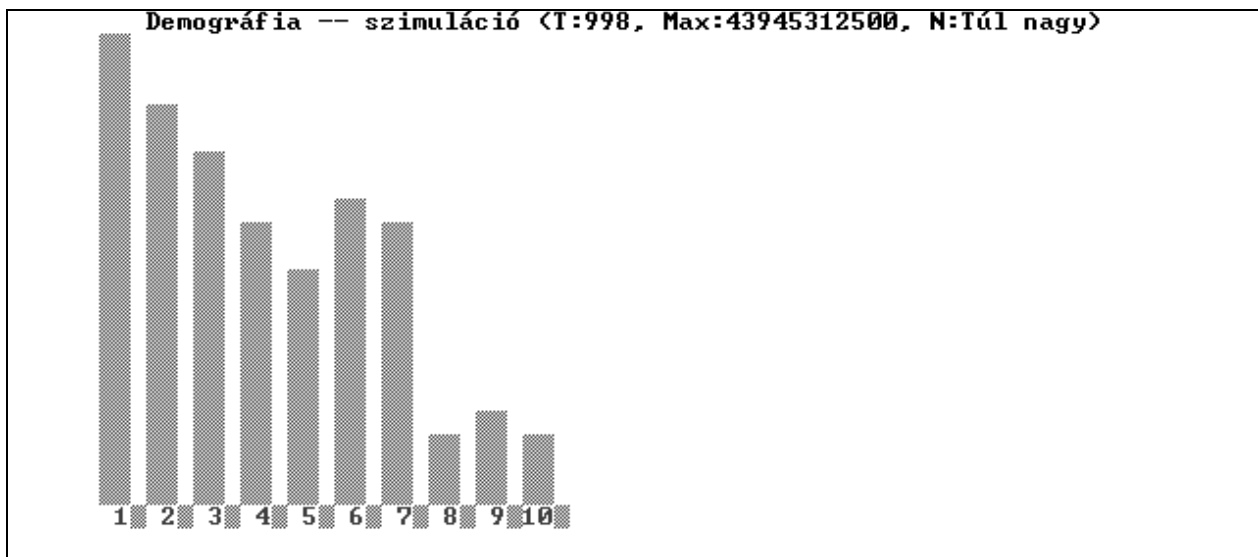
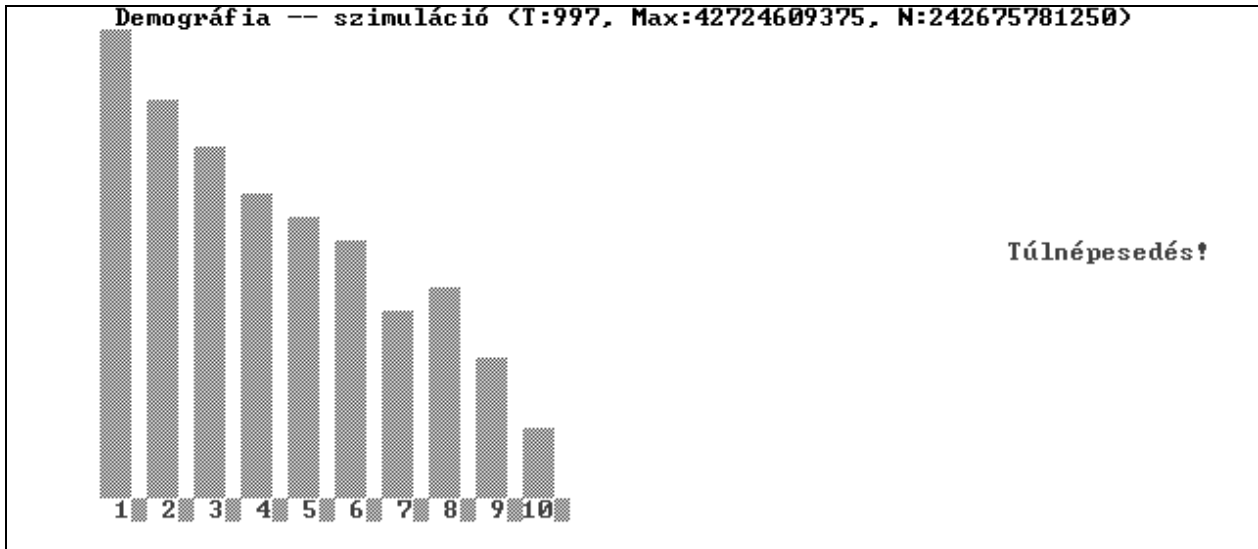
5. ábra. Egy futási kép – Egy újabb „túlnépesedési” szituáció – korcsoport-eloszlási egyensúly közelében.

Az **egyensúlyi eloszlást** akkor közelíti meg a szimuláció, amikor már keveset változik a korcsoport-eloszlás diagramja. Megfigyelhetjük, hogy még ekkor is az összlétszám folyamatos változásban (általában: növekedésben) van. Érdeemes megjegyezni: a változás trendjét!²

¹ Ezt a túlnépesedési jelenséget lehet ritkítani, ha a MaxN fordítási paraméter nagyobbra (pl 50000-re) állítjuk.

Az egyensúlyhoz közel kerülés érzékelését akár automatizálhatjuk is! E programbővítés a használó szempontjából komoly segítség lehet.

A túlnépesedési szituációt megoldó felszorzásnak is van korlátja. Ez látszik a 6. ábrán. Ez az érték, mint látjuk, kellően nagy (242 675 781 250!) ahhoz, hogy ne érezzük komoly korlátnak



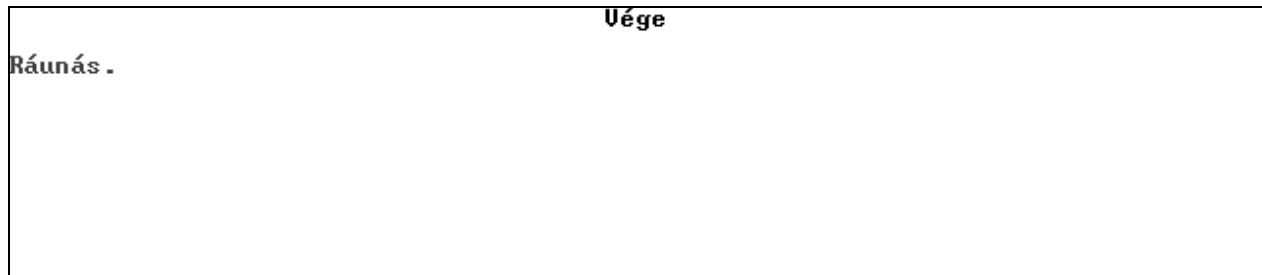
Vége

Túlnépesedés .

**6. ábra. Egy futási kép – Egy újabb „túlnépesedési” szituáció – nem sokkal a kényszerű vég előtt. (Felső kép)
 És a már túl nagy létszám miatti leállás, mint végállapot (Középső kép)
 A véget egyértelműsítő búcsú kép. (Alsó kép)**

Egy másik végkifejlet a futásban: amikor az exponenciálisan növekvő populáció korcsoport-eloszlása –megítélésünk szerint– már elérte a (dinamikus) egyensúlyi helyzetet, és ezért megállítjuk a szimulációt

² Ellenőrizze pl. egy táblázatkezelővel, hogy exponenciális ez a változási trend. Ehhez épesszű bővítés a programban: az adatok bizonyos időnként (pl. ún. CSV) fájlba kivitele.



7. ábra. Egy futási kép – Egy kilépési ok a program közvetlen megállása előtt.

A továbblépés előtt lássunk egy „in vivo” próbát! ([demogr.exe](#)) Keressünk eltérő végkimenetelű paraméter-együtteseket! Milyen csoportokba sorolhatók a paraméterek, felfedezhetők-e valamilyen szabályszerűségek?

2. A problémák

A kezelés és a látvány, valamint a szimuláció specifikus problémái és megoldásötletei.

2.1. Képernyő

A modell működésének legjellemzőbb állapotátározója a *korcsoport-eloszlás* és az *összlétszám*. Így ezek kellene, hogy minden időegységben megjelenjenek a képernyőn. Mivel a korcsoportok száma a gyakorlatban 10-20 közötti, megelégedhetünk a *konzolos* programok szokásos képernyőjével.³

2.2. Billentyűfigyelés

Ha már nem óhajtjuk a szimulációt, kiléphessünk egy billentyű lenyomásával, ill. lehessen időlegesen felfüggeszteni a futást.

2.3. Tájékoztató(k)

Egyszerű szerkezetű text-fájlból, a karakteres képernyőre, lapozottan írandó. Szükséges ismeretek: l. a korábbi „text-kezelési minimum”.

2.4. Szimulációs keret

Alapfeltételezés: az egyes korcsoportok *azonos* hosszúságú időintervallumot ölelnek föl. A szimuláció ebben az időintervallumos egységben dolgozik, azaz egy lépés során a túlélők pontosan egy korcsoporttal jutnak „előbbre”.

Alapelképzelés: minden *egyedet* tárolunk, és a *korcsoportindexével* azonosítunk.

A szimuláció során az *összlétszám* –többnyire– *exponenciálisan* nő (vagy csökken), alig található olyan paraméterezés, amely mellett az *összlétszám* stabil marad. Ez komoly fejtörést okoz algoritmikusan (is)!

Az algoritmikus megoldás ötlete: a gyermekeket nem helyezzük el rögvest a táblában, hanem csak számláljuk. Majd ellenőrizzük, hogy hozzávehetők-e az egyed-táblához a „túlsordulás” veszélye nélkül. Ha nem, akkor *korcsoportlétszám-arányos tömörítést* hajtunk végre.

³ Megjegyzem: érdemes az eddig használt szimulációs keretprogramunk felhasználásával az igényesebb felületű programot elkészíteni.

Modellezés alapadatai:

- T – idő
- N,K – egyedek és korcsoportok száma (*modell-specifikus*)
- egyedek – kora, korcsoport-sorszám (i) (*modell-specifikus*)
- szulRatak, halRatak – születési és halálozási valószínűségek korcsoportonként (*modell-specifikus*)
- kCsElo – korcsoport-eloszlás (*modell-specifikus*)
- felSzorz – a tárolt és a „feltételezett” össznépeség közötti szorzó (*modell-specifikus*)
- Tulnepesedes – kezelhető-e még a létszám (*modell-specifikus*)

A nagyvonalú algoritmus (Pascal-ul):

```
{inicializálás:}4 Param;
T:=0;
{KezdőKépernyő + EredmenyMegjelenites:} KorCsopKi;
While {not VegeE:} Folytatas and N>0 and not Tulnepesedes do
Begin
  SzimulaciosLepes;
  Inc (T);
  {EredmenyMegjelenites:} KorCsopKi;
End;
{OsszefoglaloMegjelenites:}
If
  N=0 then Write (CrLf+'Kihalás.') else if
  Tulnepesedes then Write (CrLf+'Túlnépesedés.')
  else Write (CrLf+'Ráunás.')
{EndIf};
```

Az eljárások/függvények rövid leírása:

- Param:
 - Paraméterbeolvasás – a nagyszámú paraméterhez megoldandó az igény szerinti fájlból olvasás (*modell-specifikus*)
 - Eseménytér inicializálása – kezdő populáció generálása
- KorCsopKi:
 - Aktuális idő kiírása
 - Képernyőre normálás
 - Korcsoport gyakoriság oszlopainak kirajzolása
- Folytatas „és környéke”:
 - Billentyűfigyelés –üzemmódtól függően: várakozás–
 - Végállapot-figyelés (*modell-specifikus*)
- SzimulaciosLepes:
 - A lényeg – modellváltozók módosítása (*modell-specifikus*)
 - Az eseménytér módosítása (ha kell)
- OsszefoglaloMegjelenites:
 - Konklúzió levonása (*modell-specifikus*)

⁴ A zárójelk között megjegyzés a korábbi szimulációs (keret) programban „szokásossá” vált tevékenységre utal.

2.5. Szimulációs lépés

Mivel –tudjuk– az adott időpontban születő gyerekek lehet, hogy nem férnek be az egyed tömbbe, ezért a születésükkor nem helyezük be rögvest a tömbbe, hanem csak számláljuk őket. Amint ismert lesz a frissen születettek (1. korcsoportbeliek) száma, akkor már tudni fogjuk, hogy minden további nélkül beilleszthetők-e, vagy előtte még helyet kell számukra csinálni (korcsoport arányokat megtartó tömörítéssel).

```

gyDb:=0 [a születő gyerekek száma]
Ciklus i=1-től N-ig
  Ha Random<halRatak(egyedek(i)) akkor [meghal]
    egyedek(i):=0
  különben
    Ha Random<szulRatak(egyedek(i)) akkor [szül→gyerekszámolás]
      gyDb:+1
    Elágazás vége
    egyedek(i):+1 [öregszik]
  Elágazás vége
Ciklus vége
[hullatakarítás és számlálás:]
Hullatakarítás(N)
Ciklus amíg N+gyDb≥MaxN [túlnépesedés]
  ArányosTömörítés
  [a gyerekek szelektálása:]
  gyDB:=Kerekít((1-szelRata)*gyDb)
Ciklus vége
Ciklus i=1-től gyDb-ig
  egyedek(N+i):=1
Ciklus vége
kCsElo(1):=gyDb
N:=N+gyDb

```

3. A keretprogram

Letöltheti az alábbi kódot ([DemogKer.pas](#)). Pirossal szedett részek bizonyos szempontból érdekesek számunkra:

- a 6-11, 13-17 sorok a feltételes fordításra példa (a Turbo és a Free Pascal közüli választáshoz);
- az 54-59, 62, 69-73 (és hasonló) a fájlból és a CONsolról történő egyöntetű beolvasás megvalósítását szemlélteti;
- 160, 166-168 fontos belsőparamétert definiál, ill. eljárást specifikál;
- 176-189 egy érdekes eljárást specifikál és implementál...

```

1 Program Demografia;
2 (*
3   A demográfiai feladat kerete.
4   *)
5 Uses
6 {$IFDEF FPC -- FreePascal}

```

```

7      Crt;
8      {$ELSE -- TurboPascal}
9      Newdelay,Crt;
10     {$ENDIF}
11     {$i Altrutin.inc}
12     Const
13     {$IFDEF FPC -- FreePascal}
14     MaxN=50000;
15     {$ELSE -- TurboPascal}
16     MaxN=5000;
17     {$ENDIF}
18     MaxKorcsoport=50;
19     Eps=1.0E-9;
20     bEsc=#27;
21     bSpace=' ';
22     bTeli='█';
23     Type
24     TEgyed=Byte;           {korcsoport-sorszám}
25     TEgyedSzam=LongInt;
26     TValoszinuseg=Real; {0..1}
27
28     TEgyedIndex= 0..MaxN;
29     TEgyedek=Array [TEgyedIndex] of TEgyed;
30     TKorcsoportIndex=1..MaxKorcsoport;
31
32     TValosKCs=Array [TKorcsoportIndex] of TValoszinuseg;
33     TEgeszKCs=Array [TKorcsoportIndex] of TEgyedSzam;
34     Var
35     N:TEgyedIndex;
36     K:TKorcsoportIndex;
37     egyedek:TEgyedek;
38     szulRatak,
39     halRatak:TValosKCs;
40     kCsElo:TEgeszKCs;
41     T:LongInt;
42     leptek,           {a képernyőre normáláshoz}
43     felSzorz:Real; {a tároltból a tényleges létszám meghatározásához}
44     Tulnepesedes:Boolean;
45
46     Procedure Param;
47         Var
48             i:TEgyedIndex;
49             fNBe:String;
50             fBe:Text;
51     Begin
52         UjLap('Demográfia -- paraméterek beolvasása',0);
53         Write(CrLf,'Fájlnev (ha "", akkor konzol):'); Readln(fNBe);
54         Assign(fBe,fNBe);
55         {$i-}Reset(fBe);{$i+}
56         If (IOResult<>0) or (fNBe='') then
57             Begin
58                 fNBe:='CON'; Assign(fBe,fNBe); Reset(fBe);
59             End;
60         {Most nem ellenőrizzük, csak bambán olvasunk, hiszen lehet, h. fájlból jön}
61         Write(CrLf,'Induló létszám:'); Readln(fBe,N);
62         If fNBe<>'CON' then Writeln(N);
63         Write(CrLf,'Korcsoport-szám:'); Readln(fBe,K);

```



```

64   If fNBe<>'CON' then Writeln(K);
65   Writeln(CrLf,'Halálzási ráták:');
66   For i:=1 to K-1 do
67   Begin
68     Readln(fBe,halRatak[i]);
69     If fNBe<>'CON' then
70     Begin
71       If (i Mod 10)=0 then Writeln;
72       Write(halRatak[i]:8:5);
73     End;
74   End;
75   If fNBe<>'CON' then Writeln;
76   halRatak[K]:=1;
77   Writeln(CrLf,'Szülési ráták:');
78   For i:=1 to K-1 do
79   Begin
80     Readln(fBe,szulRatak[i]);
81     If fNBe<>'CON' then
82     Begin
83       If (i Mod 10)=0 then Writeln;
84       Write(szulRatak[i]:8:5);
85     End;
86   End;
87   szulRatak[K]:=0; {mint kiderül, be sem kellene állítani!}
88   If fNBe<>'CON' then
89   Begin
90     Writeln; BillreVar;
91   End;
92   Close(fBe);
93   {Kezdő populáció kitöltés: mindenki 1. korcsoportos}
94   For i:=1 to N do
95   Begin
96     egyedek[i]:=1;
97   End;
98   kCsElo[1]:=N;
99   For i:=2 to K do
100  Begin
101    kCsElo[i]:=0;
102  End;
103  {egyéb belső paraméterek:}
104  leptek:=1.0; felSzorz:=1; Tulnepesedes:=False;
105 End;
106
107 Function Folytatas:Boolean;
108   Var
109     c:Char;
110 Begin
111   If KeyPressed then
112   Begin
113     c:=ReadKey; c:=ReadKey; {újabb billre tovább}
114   End
115   Else
116   Begin
117     c:=bSpace;
118   End;
119   Folytatas:=c<>bEsc
120 End;

```

```

121
122 Procedure KorCsopKi;
123   Const
124     MaxSorDb=20;
125   Var
126     i,j,max:Integer;
127     tS,maxS,nfS:String;
128 Begin
129   Str(T,tS);
130   max:=kCsElo[1];
131   For i:=2 to K do
132     Begin
133       If kCsElo[i]>max then max:=kCsElo[i]
134     End;
135     {normálás a képernyőre;}
136     If max>0 then leptek:=MaxSorDb/max;
137     If felSzorz>Eps then
138       Begin
139         Str(N/felSzorz:4:0,nfS)
140       End
141       else
142         Begin
143           nfS:='Túl nagy'; Tulnepesedes:=True;
144         End;
145       Str(max/felSzorz:4:0,maxS);
146       {kirajzolás a képernyőre;}
147       UjLap('Demográfia -- szimuláció (T:'+tS+', Max:'+maxS+', N:'+nfS+')',0);
148       For i:=1 to K do
149         Begin
150           GotoXY(4+i*3,22); Write(i:2,bTeli);
151           For j:=1 to Round(leptek*kCsElo[i]) do
152             Begin
153               GotoXY(4+i*3,22-j); Write(bTeli+bTeli);
154             End;
155           End;
156           Delay(500);
157       End;
158
159 Procedure SzimulaciosLepes;
160   Const
161     szelRata = 0.8; {a tömörítéskor alkalmazott szelekciós ráta}
162   Var
163     gyDb:TEgyedSzam;
164     i,j:TEgyedIndex;
165
166   Procedure Hullatakaritas(Var N:TEgyedIndex);
167   {hullatakarítás: 0-k kihagyása az egyedek[1..N] tömbből
168     N = tömör egyedek tömb hossza
169     számlálás: kCsElo[1..K] = korcsoportok gyakorisága}
170   Var
171     i,j:TEgyedIndex;
172   Begin
173     (* ide kell a lényeg *)
174   End;
175

```

Demografia1 / 2012.12.07.

```
176 Procedure AranyosTomorites;  
177   {Átlagosan minden x. marad meg, a többi "meghal";  
178   felSzorz = felSzorz/x  
179   most x=1/(1-szelRata)=5}  
180   Var  
181     i:TEgyedIndex;  
182 Begin  
183   GotoXY(65,11); Writeln('Túlnépesedés!'); BillreVar;  
184   For i:=1 to N do  
185     Begin  
186       If (egyedek[i]>0) and (Random<szelRata) then egyedek[i]:=0  
187     End;  
188     Hullatakaritas(N);  
189     felSzorz:=felSzorz*(1-szelRata);  
190   End;  
191  
192   Begin {Szimulacios Lepes}  
193     (* ide kell a lényeg *)  
194   End;  
195  
196 Begin  
197   UjLap('Demográfia',0);  
198   Param; T:=0;  
199   KorCsopKi;  
200   While Folytatas and (N>0) and not Tulnepesedes do  
201     Begin  
202       Inc(T);  
203       SzimulaciosLepes;  
204       KorCsopKi;  
205     End;  
206     UjLap('Vége',-1);  
207     If  
208       N=0 then Writeln(CrLf+'Kihalás.') else if  
209       Tulnepesedes then Writeln(CrLf+'Túlnépesedés.')  
210       else Writeln(CrLf+'Ráunás.')  
211     {EndIf};  
212     BillreVar;  
213 End.
```

Az alapfeladat megoldásához minden szükséges kellék helye: [Demogr Gyak1.zip](#).

(To be continued: [Demografia2.pdf](#) ☺)