

## 9 A REPEAT..UNTIL ciklus

- a **REPEAT ... UNTIL** ciklus
- véletlen számok generálása

### 9.1 A REPEAT ... UNTIL (háttesztelős ciklus)

A **while..do** ciklusnál a számítógép először megvizsgálta a feltételt és csak utána hajtotta végre a ciklusban levő parancsokat (amennyiben a feltétel igaz volt). Ha a feltétel mindjárt az első vizsgálatnál hamis volt, akkor a ciklusban levő utasításokat egyszer sem hajtotta végre.

A **repeat..until** ciklusnál a számítógép először végrehajtja a ciklusban levő parancsokat (repeat..until közötti részt), majd utána vizsgálja meg a feltételt. Ha a feltétel igaz, kilép a ciklusból. Ha a feltétel hamis, megismétli ismét a ciklusban levő parancsokat majd ismét megvizsgálja a feltételt. Ennél a ciklusnál tehát egyszer mindenképpen lefutnak a ciklusban levő parancsok.

A **repeat..until** ciklusból a számítógép a **while..do** ciklussal ellentétben akkor lép ki, ha a feltétel igaz (a **while..do** ciklusnál akkor lépett ki, ha a feltétel hamis volt).

A ciklus szerkezete:

```
repeat
  parancs1;
  parancs2;
  ...
  parancsN;
until feltétel;
```

Magyarul: ismételd a ciklusban levő parancsokat, amíg a feltétel nem lesz igaz (tehát amíg a feltétel hamis).

### 9.2 Véletlen számok generálása

A **random(n)** funkció egy véletlen számot ad vissza a <0,n-1> intervallumból.

Például:

**a := random(7);**           kigenerál egy egész számot, melynek értéke 0-tól 6-ig lehet  
**a := 3 + random(10);**   kigenerál egy egész számot, melynek értéke 3-tól 12-ig lehet

A véletlenszám generátor a program indítása után mindig ugyanonnan kezdi a számok generálását. A programunk elején ezért mindig inicializáljuk a **randomize;** paranccsal (ezt a parancsot úgy foghatjuk fel, mintha megkeverné a számokat). A **randomize;** parancsot elég csak egyszer, a program legelején megadnunk, mielőtt használjuk a **random()** funkciót.

**Feladat:** A számítógép véletlenszerűen válasszon (gondoljon) egy számot 1-től 5-ig. Kérdezze meg a felhasználótól melyik ez a szám. A felhasználó addig találgathat, amíg nem találja el ezt a számot. A számítógép csak annyit írjon neki ki: "eltaláltad, ez az a szám", vagy "sajnos nem ez az a szám".

**Megoldás:** A programunk a következő képen néz ki:

```
program Pelda24;
uses crt;
var a,b:integer;
begin
  clrscr;
  randomize; { megkeveri a számokat }
  a:=random(5)+1; { kigeneral egy egesz szamot 1-tol 5-ig }
  repeat
    write('Melyik szamra gondoltam? ');
    readln(b);
    if a=b then writeln('Eltalaltad, ez az a szam!')
      else writeln('Sajnos nem ez az a szam. ');
  until a=b;
  writeln('Nyomd meg az ENTER billentyut. ');
  readln; { a szamitogep var az ENTER megnyomasara }
  readln;
end.
```

## Feladatok:

1. Írjad ki *Repeat* ciklussal 100-13-ig a páros számokat! (5 karakter távolsággal) (**repeat01**)

100	98	96	94	92
90	88	86	84	82
80	78	76	74	72
70	68	66	64	62
60	58	56	54	52
50	48	46	44	42
40	38	36	34	32
30	28	26	24	22
20	18	16	14	

2. Készítsünk programot, amely bekér egész számokat mindaddig, amíg nem adjuk meg a 0-t. A program határozza meg és írja ki a beadott egész számok közül a legkisebbet és a legnagyobbat. (A 0-t ne számítsa bele a beadott számokba, ez csak a bevétel végét jelzi.) A számok beolvasását a 0 végjelig *repeat .. until* ciklus segítségével valósítsuk meg! (**repeat02**)

```
Adja meg a szamot: 12
Adja meg a szamot: 33
Adja meg a szamot: 7
Adja meg a szamot: 48
Adja meg a szamot: 0
-----
A legkisebb szam: 7
A legnagyobb szam: 48
```

3. Készítsünk játékprogramot, amely gondol egy számot 1 és 50 között. A felhasználó addig találgathat, amíg nem találja el a keresett számot. A számítógép minden rossz tipp után írja ki, hogy a gondolt szám nagyobb vagy kisebb. (**repeat03**)

```
Gondoltam egy szamra 1 es 50
kozott! Tipp: 25
Kisebb
Tipp: 12
Nagyobb
Tipp: 18
Eltalaltad!!! Gratulalok!
```

4. Egészítsük ki az előző programunkat úgy, hogy a játékos csak maximum 7-szer tippelhessen. Ha a hetedik tippre sem találja el a gondolt számot, a program írja ki a gondolt számot majd fejeződjön be. (**repeat04**)

```
...
7. tipp: 33
Nem talaltad el! A gondolt szam:
32.
```

5. Olvassunk be pozitív egész számokat 0 végjelig. Írjuk ki a számok átlagát. (A 0-t ne számítsa bele a beadott számokba, ez csak a bevétel végét jelzi.) (**repeat05**)

```
Adja meg a szamot: 12
Adja meg a szamot: 33
...
Adja meg a szamot: 0
A szamok atlaga: 7,321
```

6. Készítsünk programot, amely ki fogja kérdezni a matematikát (két szám összeadását, az <1,10> intervallumból). A két számot a számítógép véletlenszerűen válassza ki. A program akkor fejeződjön be, ha a felhasználó 10 példát kiszámolt helyesen. Rossz válasz esetén kérdezze újra ugyanazt a példát. A program végén írjuk ki az eredményességet százalékokban. (**repeat06**)

```
Szamolj!
1. 2+5=7
2. 8+5=13
3. 4+5=10
3. 4+5=9
...
10. 3+9=12
Eredmeny: 10-bol 9 jo megoldas
```