

04_PythonBev_vezerlesi_szerk

August 23, 2024



1 Vezérlési szerkezetek II.

- Pythonban a vezérlési szerkezetek belsejét behúzással kell jelölni.
- Ezt a behúzást a Tab billentyű lenyomásával érhetjük el

1.1 while ciklus

- lehet, hogy nem tudjuk előre a lépések számát...
- [while utasítása Python dokumentációban](#)
- Szintaxis:

```
while FELTÉTEL:  
    UTASÍTÁS
```

- Egy jól megírt program esetén **az utasítás a feltételt előbb-utóbb hamisra állítja.** (Ellenkező esetben, ha a feltétel igaz, akkor az is marad, így végtelen ciklus keletkezik.)

1.1.1 1. példa: Móricka a programozásvizsgán

Móricka addig megy a programozás vizsgára amíg az nem sikerül neki. A ponthatár 16 pont.

```
[2]: ## Példa: Móricka a programozásvizsgán.  
while int(input('Hány pontot értél el? ')) < 16:  
    print('Tanulj még!')  
  
print('Gratulálok, átmentél.')
```

Gratulálok, átmentél.

1.1.2 2. példa: Testmagasság bekérése amíg valós adatot nem kapunk

- Kérjük be a felhasználótól a centiméterben megadott testmagasságot (h)
- Vegyük figyelembe, hogy a valaha létező
 - legalacsonyabb ember 54,6 cm,
 - a legmagasabb ember pedig 272 cm-es volt!

```
[3]: # Megoldás:
h = float ( input("Adja meg a testmagasságát cm-ben: ") )

while (h < 54.6) or (h > 272):
    print("Ez nem lehet!")
    h = float ( input("Adja meg a testmagasságát cm-ben: ") )

print(f"Az ön testmagassága {h:0.1f} cm.")
```

Ez nem lehet!

Az ön testmagassága 123.0 cm.

1.1.3 3. példa

Ki tudnánk írni az első 10 negyzetszámot while segítségével a képernyőre?

```
[4]: # Megoldás:
i = 1
while (i <= 10):
    print(i**2)
    i += 1
```

1
4
9
16
25
36
49
64
81
100

1.1.4 Hibalehetőségek:

- Végtelen cilus: a feltétel mindig teljesül
- El sem induló ciklus: a feltétel már az első lépésben elbukik

```
[4]: # Végtelen ciklus ha mindig teljesül a feltétel.
# NE FUTASSA!

# i = 1
# while (i <= 10):
#     print(i**2)
```

```
[5]: ## El sem induló ciklus
i = 11
print(i)
```

```
while (i <= 10):
    print(i**2)
    i += 1
print('Vége')
```

11
Vége

1.2 Ciklus futásának befolyásolása

- break - azonnali kiugrás a ciklusból
- continue - az adott “ütem” továbbléptetése

1.2.1 1. példa: Móricka a programozásvizsgán (break verzió)

```
[6]: # break használatával
while True:
    pont = int(input('Hány pontot értél el? '))
    if pont < 16:
        print('Tanulj még!')
    else:
        print('Gratulálok, átmentél.')
        break
```

Gratulálok, átmentél.

1.2.2 2. példa: Testmagasság bekérése (break verzió)

```
[7]: # break használatával
while True:
    h = float ( input("Adja meg a testmagasságát cm-ben: ") )
    if (h < 54.6) or (h > 272):
        print("Ez nem lehet!")
    else:
        print(f"Az ön testmagassága {h:0.1f} cm.")
        break
```

Az ön testmagassága 123.0 cm.

1.2.3 3. példa: Páros négyzetszámok kiírása 100-ig

```
[8]: # range testreszbálásával:
N = 10
for elem in range(2, N+1 ,2):
    print(elem**2)
```

4
16

36
64
100

```
[9]: # if segítségével párosságot ellenőrizve
N = 10
for elem in range(1, N + 1):
    if elem%2 == 0:
        print(elem**2)
```

4
16
36
64
100

```
[10]: # if segítségével páratlanságot ellenőrizve és ütemet átlépve
N = 10
for elem in range(1, N + 1):
    if elem%2 != 0:
        continue
    print(elem**2)
```

4
16
36
64
100

1.3 Gyakorlás: Egyszerű számkitalálós játék

Készítsünk programot, amely sorsol egy egész számot 1-től 100-ig, majd tippeket kér a játékostól, amíg a játékos el nem találja a számot. A program minden tipp után írja ki, hogy a megadott tipp túl kicsi, túl nagy vagy helyes volt-e!

```
[12]: # Az (ál)véletlenszám-generáló modul importálása.
import random

# Véletlen egész szám kisorsolása 1 és 100 között. -- randint() függvény a
↳ random modulból
n = random.randint(1,100)

# Tippek kérése, amíg a játékos el nem találja a számot.

tipp = -1
while n != tipp:
    # ---- tipp bekérése ----
    tipp = int(input("Adja meg a tippet: "))
```

```
#-----  
  
if tipp > n:  
    print("Ez túl nagy tipp.")  
elif tipp < n:  
    print("Ez túl kicsi tipp.")  
else:  
    print("Gratulálunk, eltalálta!")
```

```
Ez túl kicsi tipp.  
Ez túl kicsi tipp.  
Ez túl kicsi tipp.  
Ez túl kicsi tipp.  
Ez túl nagy tipp.  
Ez túl nagy tipp.  
Ez túl nagy tipp.  
Ez túl nagy tipp.  
Ez túl nagy tipp.  
Ez túl nagy tipp.  
Gratulálunk, eltalálta!
```

```
[ ]: # randint függvény dokumentációs sztringje!  
random.randint.__doc__
```

```
[ ]: 'Return random integer in range [a, b], including both end points.\n'
```