

17.) LISTÁK AZ ELJÁRÁSOKBAN ÉS FÜGGVÉNYEKBE

A függvényeknél megtanultuk, hogy egy értéket adnak vissza a főprogramnak. **A python nyelv lehetőséget ad arra, hogy az érték az összetett legyen, mondjuk egy lista.** Tehát át is tud venni egy listát és vissza is tud adni egy listát.

A cím (referencia) szerinti paraméteradásnál **a főprogram és az alprogram a változó nevéhez ugyanazt a memória területet rendeli hozzá.** Tehát **használhatunk két nevet ugyanannak a listának.**

- A jobb oldali szövegdobozban látszik, hogy a 3. sorban kezdődik a végrehajtás, mert az a főprogram első sora.
- Az „a” nevű változóba az [1,2,3] listát tesszük.
- A 4. sorban meghívjuk a proced nevű eljárást, az „a” változót használva paraméterül.
- A program végrehajtása az 1. sorban folytatódik, ahol az „a” változó memóriaterületéhez fog csatlakozni a most létrejövő „n” változó, ami a korábban létrehozott listát tartalmazza.
- A 2. sorban hozzáfűzünk az „a” listához egy új elemet, a 4-et. Mivel csak egy listánk van két névvel, ezért az „a” listánk is módosul.
- A program végrehajtása, a főprogram 6. sorába ugrik, ahol kiíratjuk az „a” változót. Mivel az „a” változó ugyanaz mint az „n” változó, csak, ezért az eredmény a [1,2,3,4] lista.

Nézzünk egy példát:

```

1 def proced(n)
2     n.append(4)
3 a=[1,2,3]
4 proced(a)
5 print(a)
    
```

0
1
2
3
4



Tehát ha az eljárás vagy függvény paramétere összetett változó, például lista, az átadás referencia szerinti, így a főprogramban található változót az alprogram módosíthatja. Ha a változó egyszerű változó, vagy konkrét érték, az átadás érték szerinti, tehát a főprogramban található változó nem módosul. Ha egyszerre adunk át listát és egyszerű változót, akkor mindkét átadás referencia szerinti lesz. A karakterlánc egyszerű változónak számít.

(17a.py)

Készítsünk egy programot, melyben feltöltünk egy listát véletlenszerű „X” és „O” karakterekkel!

- A feladat megoldásához használjunk eljárást és függvényt!
- Az első sorban elérhetővé tesszük a véletlenszámok generálását.
- Majd egy megjelenítést lehetővé tevő eljárást készítünk „kiiratas” néven. (3-5. sor)
- A szokásos módon def utasítással kezdünk majd megadjuk az eljárás nevét és zárójelben az átvett lista nevét és a kettőspontot a végére.
- For utasítással végig megyünk a listán és kiíratjuk az elemeket, szóközzel elválasztva. Majd egy sortöréssel zárjuk az eljárást.
- Aztán elkészítjük az adatokat elkészítő modult 8-16. sorban. Ebben az esetben egy függvényt hozunk létre.
- A „generalas” nevű függvényben véletlenszám generálás segítségével a minta szerint feltöltjük a listánkat!
- A program fő részében (18-20. sor) már csak létrehozunk egy üres listát, amit a következő sorban feltöltünk a függvény meghívásával!
- Végül a kiírató eljárás meghívásával kiíratjuk az feltöltött listát!

```

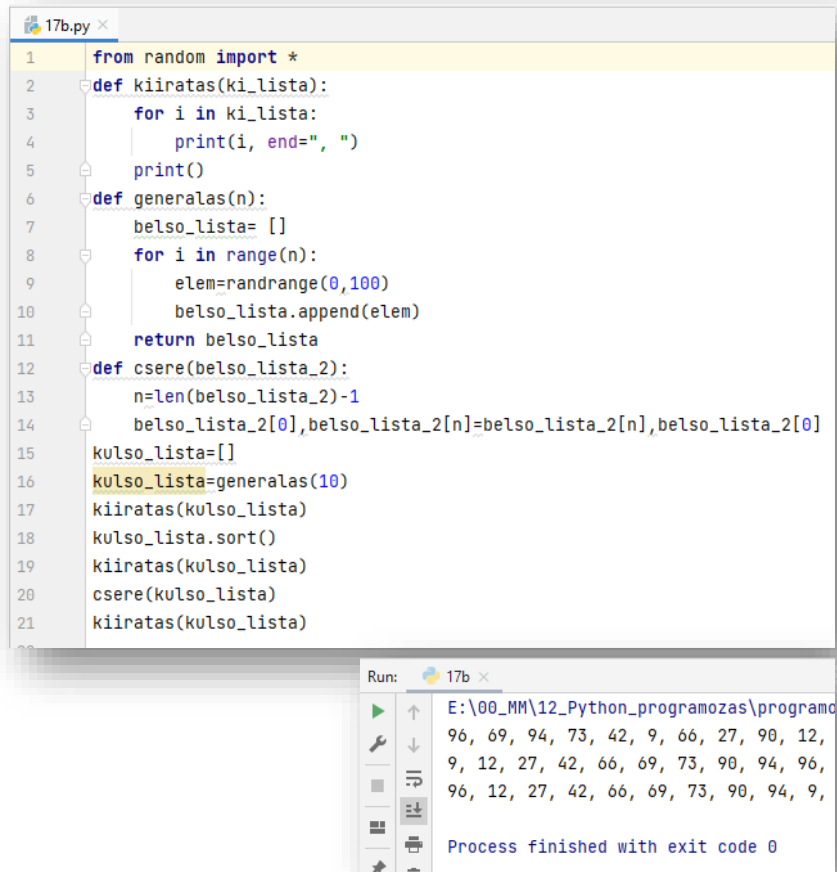
17a.py x
1 from random import *
2
3 def kiiratas(ki_lista):
4     for i in ki_lista:
5         print(i,end=' ')
6     print()
7
8 def generalas(c):
9     gen_lista=[]
10    for i in range(c):
11        elem=randrange(0,2)
12        if elem % 2 ==0:
13            gen_lista.append("X")
14        else:
15            gen_lista.append("O")
16    return gen_lista
17
18 fo_lista=[]
19 fo_lista= generalas(10)
20 kiiratas(fo_lista)
    
```

Run: 17a x

C:\Users\eloado\PycharmPro
O X O O O X O X X X

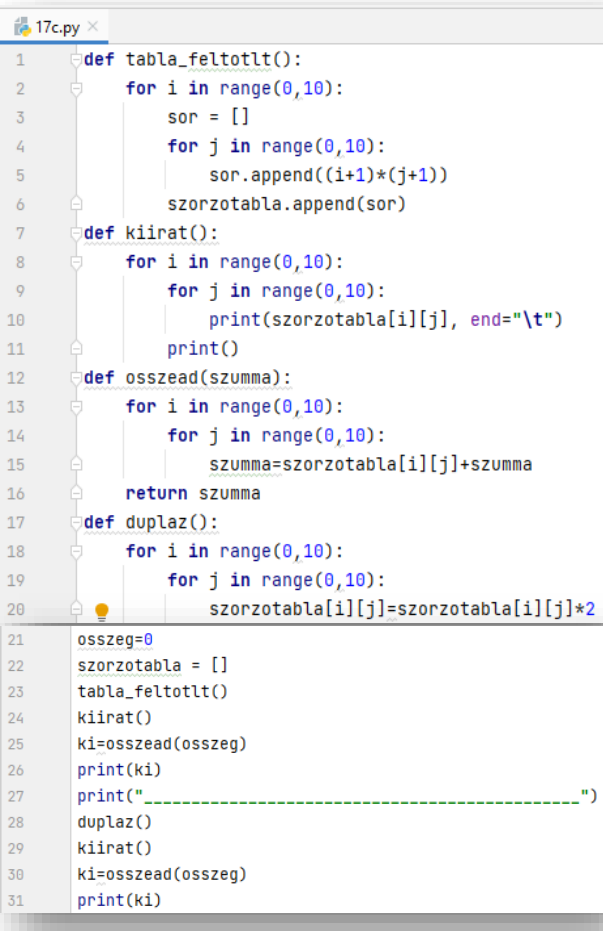
(17b.py)

Készítsünk programot, melyben generálunk egy 10 elemű listát, amiben 0 és 100 közötti véletlen számokat helyezünk el!
 Kiíratjuk ezt a listát az első sorba!
 Aztán rendezzük növekvő sorrendbe ezt a listát és azt is kiíratjuk!
 Majd a lista első és utolsó elemét cseréljük meg, végül ezt is kiíratjuk a harmadik sorba!
 A feladat megoldásánál alkalmazzunk eljárásokat és függvényeket!



Először értelmezzük a programot a képernyőkép alapján!
 Majd gépeljük be és teszteljük!

(17c.py)



Készítsünk programot, melyben listákkal dolgozunk eljárásokban és függvényekben!

- töltünk fel egy 2D tömböt szorzótáblával
- írassuk ki
- adjuk össze a szorzótábla értékeit
- duplázunk meg a szorzótábla elemeinek értékét

Értelmezzük, gépeljük be, futtassuk a programot!

