

PYTHON TEST

1. Python este un limbaj compilat/interpretat, tipat static/dinamic, multiparadigma.
2. Explicati ce inseamna tipare statica/dinamica, in functie de raspunsul dat.
3. Ce afiseaza urmatorul cod rulat intr-un mediu python3:

```
v1 = { I : [ j for j in range(i) ] for i in range(4) }  
print(v1)
```

4. Implementati in python3 functia zipWith folosind list comprehension.

zipWith :: (a -> b -> c) -> [a] -> [b] -> [c]

zipWith(functie, lista_1, lista_2)

exemplu de folosire:

```
import operator  
zipWith(operator.add, range(0,5), range(10, 20))  
[10, 12, 14, 16, 18]
```

Hint: puteti folosi functia zip, care primeste 2 liste si returneaza o lista de perechi cu elementele de pe aceeasi pozitie

```
list(zip([1,2,3], [4, 5, 6]))  
[(1, 4), (2, 5), (3, 6)]
```

5. Implementati in python3 functia fileWalker. Puteti sa utilizati functii din modulul os, cu exceptia os.walk.

fileWalker(cale_de_inceput)

Functia trebuie sa afiseze calea catre toate fisierele continute in directorul identificat prin cale_de_inceput si in toate subdirectoarele acestuia.

Se recomanda folosirea urmatoarelor functii de modului os:

```
os.listdir(path)  
os.path.join(path1, path2)  
os.path.isdir(path)  
exemplu de folosire:
```

Fie directorul tmp cu urmatoarea structura (nume/ inseamna director) :

```
tmp/  
  a  
  b  
  c/  
    aa  
    bb/  
      aaa  
  d  
  e/  
    aa  
    bb  
    cc
```

fileWalker(tmp) afiseaza

```
t/a
t/d
t/e/bb
t/e/aa
t/e/cc
t/c/bb/aaa
t/c/aa
t/b
```

Bonus: Ignorati exceptiile de tipul permission denied, cauzate de fisierele la care nu are permsiune de acces userul care a executat scriptul python.

6. Ce afiseaza urmatorul cod intr-un mediu python3:

```
def f(*args, **kwargs):
    print(args, kwargs)

d = dict(zip(list(map(chr, range(97, 123))), [1,2,3]))

f(1, 2, "alpha", "beta", **d)
```

7. Fie urmatorul cod:

```
import math
pi = math.pi
```

Folosind functia print, variabila pi si metoda format a unui string afisati urmatoarele mesaje:

Pi rotunjit cu 2 zecimale este 3.14

Pi convertit la un numar natural este 3_____

(Obs: in acest exemplu 3 este urmat de 6 caractere '_')

8. Fie urmatorul cod:

```
class MyClass(object):

    def __init__():
        self.a = 1
        self.b = 2
        self.c = None
```

Linia x = MyClass() genereaza urmatoarea eroare:

TypeError: __init__() takes 0 positional arguments but 1 was given

a) Corectati codul anterior

b) Adaugati metoda/metodele necesara/necesare clasei MyClass pentru a obtine urmatorul comportament:

```
l = [MyClass() for i in range(3)]
```

```
print(l) sa afiseze [1 2 None, 1 2 None, 1 2 None]
```