

# Exerciții - Săptămâna 2

## Exercițiul 1: valori distincte

Scrieți o funcție cu trei parametri (de același tip oarecare), care returnează câte valori distincte există între argumentele primite (unul, două sau trei) și tipărește, după caz, un mesaj: "toate argumentele sunt distincte/egale" sau "argumentele 1 și 2 (resp. 2 și 3, sau 1 și 3) sunt egale". Evitați pe cât posibil duplicarea de cod: pentru porțiuni de cod similare, creați (și apelați) o funcție care conține partea comună și are ca parametri valorile care diferă.

## Exercițiul 2: mediana

Scrieți o funcție care calculează mediana a trei valori (valoarea aflată între celelalte două). Încercați să scrieți cod cât mai simplu, și să nu-l repetați. Puteți folosi o funcție auxiliară care calculează mediana a trei numere, pentru care știm că primul e mai mic sau egal decât al doilea. Sau puteți încerca să compuneți doar funcțiile standard max/min de două elemente (expresia trebuie să fie oarecum simetrică). Care din variante necesită mai puține comparații?

## Exercițiul 3: operații cu funcții

În matematică, am extins uneori operatorul + de la numere la funcții, definind funcția  $f + g$  prin relația  $(f + g)(x) = f(x) + g(x)$

a) Definiți o funcție care ia ca parametru două funcții  $f$  și  $g$  și returnează funcția definită ca suma lor prin relația de mai sus.

b) Scrieți o funcție mai generală, care primește ca parametru și operatorul binar (o funcție de două argumente) care e aplicată celor două funcții. Verificați că o puteți folosi cu `operator.add` și `operator.sub` pentru a calcula suma și diferența, dar încercați și alți operatori.

## Exercițiul 4: funcție anonimă

Scrieți o funcție anonimă care adaugă 15 la un număr dat ca argument.

## Exercițiul extra:

Întotdeauna când scriem cod există posibilitatea să facem greșeli. Dacă în momentul în care compilăm sau rulăm avem erori, nu trebuie să ne panicăm. Este foarte important să putem remedia greșelile. Acest exercițiu conține mai multe secvențe de cod cu greșeli în ele. Identificați greșelile și remediați-le.

```
=====
def f x:
    return x + 1
print(f(1))
=====
def f(x)
    return x + 1;
print(f(1))
```

```

=====
def f(x):
return x + 1;
print(f(1))
=====
def f(x):
    if (x == 0):
        return true;
    else:
        return false
print(f(1))
=====
def f(x):
    if (x == 0):
        return true
    else
        return false;
print(f(1))
=====
def f(x):
    if (x == 0)
        return True
    else
        return False;
print(f(1))
=====
def f(x):
    if x == 0:
        return True
    else
        return False;
print(f(1))
=====
def f(x):
    if x == 0:
        return True
    else:
        return False;
print(f(1))
=====
def 3f():
    print("Fast food");
=====
def (x):
    return x + 1
=====
def x:
    return x + 1
=====
def increment():
    return x + 1
print(increment())
=====
def increment x:
    return x + 1

```

```
print(increment(5))
=====
increment(x):
    return x + 1
print(increment(5))
=====
import math as m
math.floor(3)
=====
import math
floor(3)
=====
def impartire(x, y):
    return x \ y;
print(f(1, 2))
=====
dof f(x):
    return x + 5
print(f(1, 2))
=====
```