****

**MINISTERUL EDUCAȚIEI, CULTURII ȘI CERCETĂRII**

**AL REPUBLICII MOLDOVA**

**Universitatea Tehnică a Moldovei**

**Facultatea Calculatoare, Informatică şi Microelectronică**

**Departamentul Informatică şi Ingineria Sistemelor**

**Calancea Catalin**

**MI-222**

**Raport**

**pentru lucrarea de laborator Nr.1**

***la cursul de “Programare Interactivă”***

Verificat:

lector universitar

Strună Vadim

**Chișinău – 20****23**

**CUPRINS**

[1.INTRODUCERE 3](#_Toc149559369)

[2.SARCINA 4](#_Toc149559370)

[3.SCRIPT 7](#_Toc149559371)

[4.CONCLUZIA 12](#_Toc149559372)

[4.WEBOGRAFIE 13](#_Toc149559373)

# 1.INTRODUCERE

Python este un limbaj de programare extrem de popular și versatil, cunoscut pentru simplitatea sa și sintaxa ușor de înțeles. Cu o comunitate activă și o serie de biblioteci și framework-uri puternice, Python este o alegere excelentă pentru programatori de toate nivelele de experiență. În această introducere, vom explora câteva dintre conceptele de bază ale limbajului Python și modul în care puteți începe să lucrați cu el.

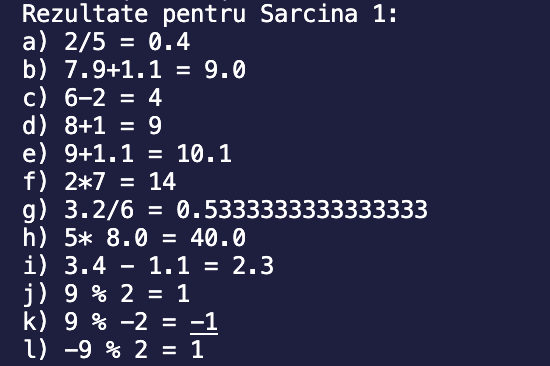
# 2.SARCINA

**I. Rezolvați sarcinile, unde este necesar faceți explicații**

1.Să se execute în consola Python toate expresiile, să se explice rezultatele:

a) 2/5  
b) 7.9+1.1 c) 6-2  
d) 8+1  
e) 9+1.1 f) 2\*7  
g) 3.2/6

h) 5\*8.0   
i) 3.4-1.1   
j) 9%2   
k) 9%-2   
l) -9%2



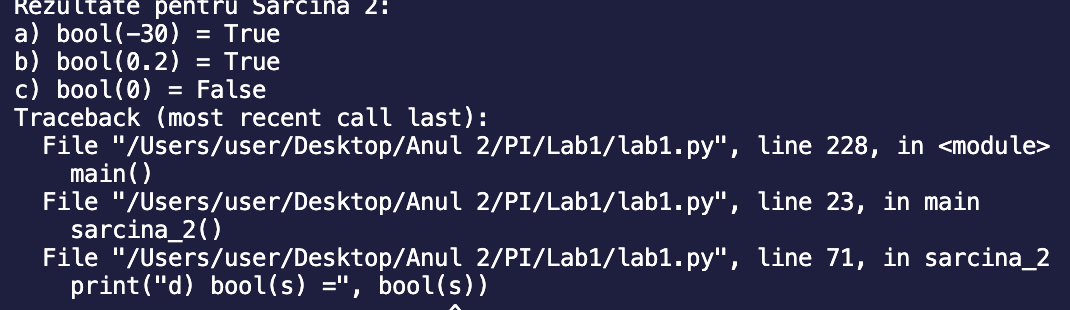
2.Sa se afișeze rezultatul executării următorului cod:

a) bool(-30)

b) bool(0.2)

c) bool(0)

d) bool(s)

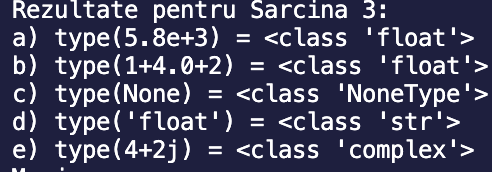


3.Să se verifice tipul și valoarea următoarei expresii:

a) type(5.8e+3)  
b) type(1+4.0+2)  
c) type(None)

d)  type(‘float’)

e)  type(4+2j)



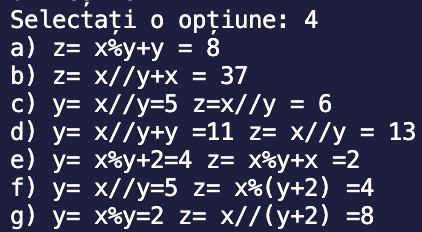
4.Să se calculeze valoarea variabilei întregi z pentru x=32 și y=6.

a) z= x%y+y

b) z= x//y+x  
c) y= x//y z=x//y  
d) y= x//y+y z= x//y  
e) y= x%y+2 z= x%y+x

f) y= x//y z= x%(y+2)

g) y= x%y z= x//(y+2)

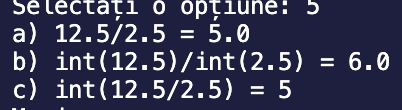


5. Să se compare și să se explice valorile expresiilor obținute:

a) 12.5/2.5

b) int(12.5)/int(2.5)

c) int(12.5/2.5)

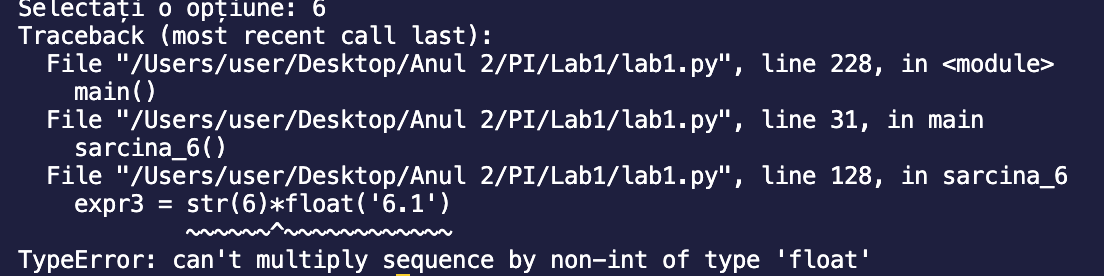


6. Să se anticipeze rezultatul fiecăreia dintre următoarele instrucțiuni, apoi să fie verificate în interpretorul Python.

a) str(6)\* int(‘5’)  
b) int(”6”)+ float(”6.1”)

c) str(6) \* float(”6.1”)

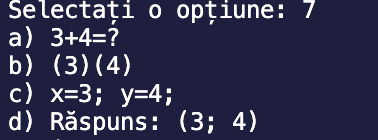
d) str(6/4) \* 2



7. Să se scrie o instrucțiune care ***afișează*** valorile variabilelor x=3 și y =4 în următorul format.

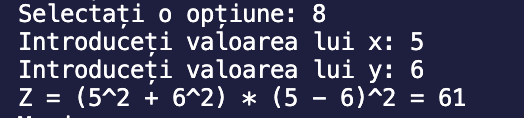
a) 3+4=?

b) (3)(4)  
c) x=3; y=4;  
d) Răspuns: (3; 4)

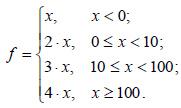


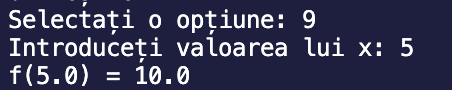
8.Să se scrie un program care ia de la tastatură două valori întregi ***x*** și ***y***, și calculează valoarea lui ***z***:

**Z=(X2 +Y2)(X-Y)2**

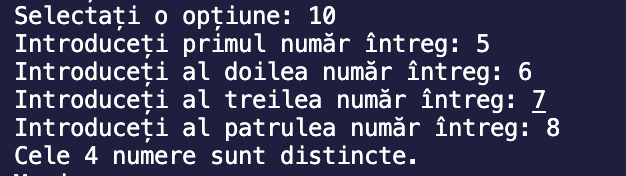


9. Se citeşte x, un număr real. Să se evalueze expresia:

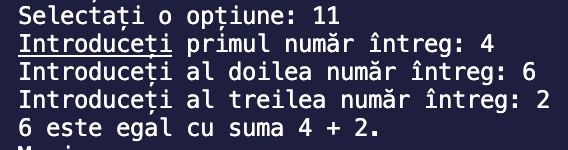




10. Se citesc 4 numere întregi. Să se decidă dacă sunt distincte (adică nu există două egale între ele).  
Indicație. Se compară primul număr cu toate celelalte, al doilea cu cele care îi urmează, al treilea cu al patrulea. Dacă în nici o comparație nu rezultă egalitate, se tipărește mesajul corespunzător.



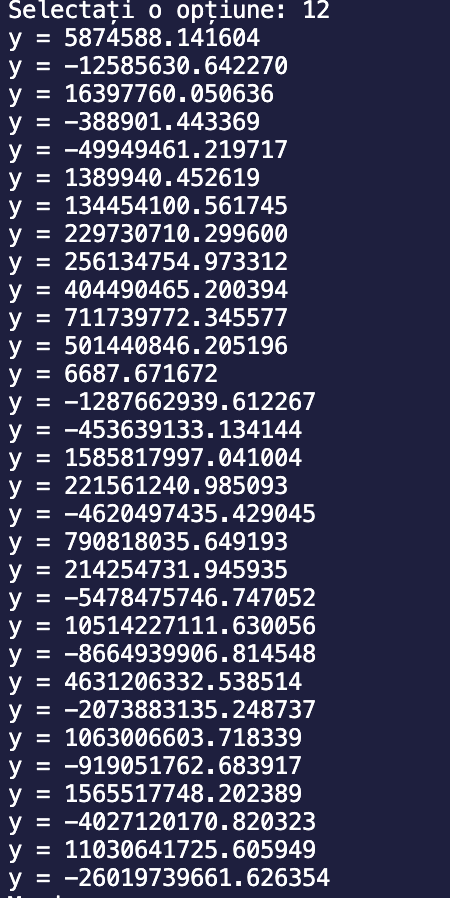
11. Se citesc 3 numere întregi. Să se tipărească, dacă există, numărul care este egal cu suma celorlalte două.



**II. Variantele sarcinilor individuale:**

Să se rezolve următoarea formulă.

|  |  |
| --- | --- |
| q=│√**d2**+f – f**2**ln**3**(d+x)/d│; y=cos**3**q**4** – q/√**f2**+d**2** | d=2.359 |
| f=15.5 |
| x=-0.1...2.9; cu pasul 0.1 |



# 3.SCRIPT

def main():

while True:

print("Meniu:")

print("1. Rezolvare expresii matematice")

print("2. Evaluare expresii booleane")

print("3. Verificare tip și valoare expresii")

print("4. Calcul valoare variabilă întreagă z")

print("5. Comparare și explicație expresii")

print("6. Anticipare și verificare expresii")

print("7. Afișare variabile x și y")

print("8. Calculul lui z cu formula dată")

print("9. Evaluează expresia f(x)")

print("10. Verifică dacă 4 numere sunt distincte")

print("11. Găsește numărul egal cu suma celorlalte două")

print("12.Individual")

print("0. Ieșire")

optiune = input("Selectați o opțiune: ")

if optiune == "1":

sarcina\_1()

elif optiune == "2":

sarcina\_2()

elif optiune == "3":

sarcina\_3()

elif optiune == "4":

sarcina\_4()

elif optiune == "5":

sarcina\_5()

elif optiune == "6":

sarcina\_6()

elif optiune == "7":

sarcina\_7()

elif optiune == "8":

sarcina\_8()

elif optiune == "9":

sarcina\_9()

elif optiune == "10":

sarcina\_10()

elif optiune == "11":

sarcina\_11()

elif optiune == "12":

individual()

elif optiune == "0":

break

else:

print("Opțiune invalidă. Reîncercați.")

def sarcina\_1():

*# Sarcina 1*

print("Rezultate pentru Sarcina 1:")

print("a) 2/5 =", 2/5)

print("b) 7.9+1.1 =", 7.9 + 1.1)

print("c) 6-2 =", 6 - 2)

print("d) 8+1 =", 8 + 1)

print("e) 9+1.1 =", 9 + 1.1)

print("f) 2\*7 =", 2 \* 7)

print("g) 3.2/6 =", 3.2 / 6)

print("h) 5\* 8.0 =", 5\* 8.0)

print("i) 3.4 - 1.1 =", 3.4 - 1.1)

print("j) 9 % 2 =", 9 % 2)

print("k) 9 % -2 =", 9 % -2)

print("l) -9 % 2 =", -9 % 2)

def sarcina\_2():

*# Sarcina 2*

print("Rezultate pentru Sarcina 2:")

print("a) bool(-30) =", bool(-30))

print("b) bool(0.2) =", bool(0.2))

print("c) bool(0) =", bool(0))

print("d) bool(s) =", bool(s))

def sarcina\_3():

*# Sarcina 3*

print("Rezultate pentru Sarcina 3:")

print("a) type(5.8e+3) =", type(5.8e+3))

print("b) type(1+4.0+2) =", type(1+4.0+2))

print("c) type(None) =", type(None))

print("d) type('float') =", type('float'))

print("e) type(4+2j) =", type(4+2j))

def sarcina\_4():

*# Sarcina 4*

x = 32

y = 6

*#a*

z1 = x % y + y

*#b*

z2 = x // y + x

*#c*

y1 = x//y

z3 = x // y1

*#d*

y2 = x // y + y

z4 = x // y2 + y2

*#e*

y3 = x % y +2

z5 = x % y3 + 2

*#f*

y4 = x//y

z6 = x % (y4 + 2)

*#g*

y5 = x % y

z8 = x // (y5 + 2)

print(f"a) z= x%y+y = {z1}")

print(f"b) z= x//y+x = {z2}")

print(f"c) y= x//y={y1} z=x//y = {z3}")

print(f"d) y= x//y+y ={y2} z= x//y = {z4}")

print(f"e) y= x%y+2={y3} z= x%y+x ={z5}")

print(f"f) y= x//y={y4} z= x%(y+2) ={z6}")

print(f"g) y= x%y={y5} z= x//(y+2) ={z8}")

def sarcina\_5():

*# Sarcina 5*

expr1 = 12.5 / 2.5

expr2 = int(12.5) / int(2.5)

expr3 = int(12.5 / 2.5)

print(f"a) 12.5/2.5 = {expr1}")

print(f"b) int(12.5)/int(2.5) = {expr2}")

print(f"c) int(12.5/2.5) = {expr3}")

def sarcina\_6():

*# Sarcina 6*

expr1 = str(6) \* int('5')

expr2 = int('6') + float('6.1')

expr3 = str(6) \* float('6.1')

expr4 = str(6/4) \* 2

print(f"a) str(6)\*int('5') = {expr1}")

print(f"b) int('6')+float('6.1') = {expr2}")

print(f"c) str(6)\*float('6.1') = {expr3}")

print(f"d) str(6/4)\*2 = {expr4}")

def sarcina\_7():

*# Sarcina 7*

x = 3

y = 4

print(f"a) {x}+{y}=?")

print(f"b) ({x})({y})")

print(f"c) x={x}; y={y};")

print(f"d) Răspuns: ({x}; {y})")

def sarcina\_8():

*# Sarcina 8*

x = int(input("Introduceți valoarea lui x: "))

y = int(input("Introduceți valoarea lui y: "))

z = (x\*\*2 + y\*\*2) \* (x - y)\*\*2

print(f"Z = ({x}^2 + {y}^2) \* ({x} - {y})^2 = {z}")

def sarcina\_9():

*# Sarcina 9*

x = float(input("Introduceți valoarea lui x: "))

if x < 0:

f = x

elif 0 <= x < 10:

f = 2 \* x

elif 10 <= x < 100:

f = 3 \* x

else:

f = 4 \* x

print(f"f({x}) = {f}")

def sarcina\_10():

*# Sarcina 10*

numar1 = int(input("Introduceți primul număr întreg: "))

numar2 = int(input("Introduceți al doilea număr întreg: "))

numar3 = int(input("Introduceți al treilea număr întreg: "))

numar4 = int(input("Introduceți al patrulea număr întreg: "))

sunt\_distincte = True

if numar1 == numar2 or numar1 == numar3 or numar1 == numar4:

sunt\_distincte = False

elif numar2 == numar3 or numar2 == numar4:

sunt\_distincte = False

elif numar3 == numar4:

sunt\_distincte = False

*# Afisăm rezultatul*

if sunt\_distincte:

print("Cele 4 numere sunt distincte.")

else:

print("Cele 4 numere NU sunt distincte.")

def sarcina\_11():

*# Sarcina 11*

*# Citim cele 3 numere întregi*

nr1 = int(input("Introduceți primul număr întreg: "))

nr2 = int(input("Introduceți al doilea număr întreg: "))

nr3 = int(input("Introduceți al treilea număr întreg: "))

*# Verificăm dacă există un număr egal cu suma celorlalte două*

if nr1 == nr2 + nr3:

print(f"{nr1} este egal cu suma {nr2} + {nr3}.")

elif nr2 == nr1 + nr3:

print(f"{nr2} este egal cu suma {nr1} + {nr3}.")

elif nr3 == nr1 + nr2:

print(f"{nr3} este egal cu suma {nr1} + {nr2}.")

else:

print("Nu există niciun număr care să fie egal cu suma celorlalte două.")

def individual():

*# Sarcina individuala*

import math

d = 2.359

f = 15.5

results = []

for x in range(-1, 30):

x /= 10.0 *# Convert x to a float with one decimal place*

q = abs(math.sqrt(d\*\*2 + f) - f\*\*2 \* math.log(d + x)\*\*3 / d)

y = math.cos(q)\*\*3 \* q\*\*4 - q / math.sqrt(f\*\*2 + d\*\*2)

results.append((x, y))

for x, y in results:

print(f"y = {y:.6f}")

if \_\_name\_\_ == "\_\_main\_\_": *#script este executat ca program principal*

main()

# 4.CONCLUZIA

În cadrul acestei lucrări de laborator, am avut oportunitatea de a ne familiariza cu mediul de programare Python și de a învăța elementele de bază ale limbajului. Am înțeles importanța instalării unui interpretor sau IDE pentru a putea crea și executa programe Python. Am explorat interfața și funcționalitatea Python IDLE, care ne-a oferit un mediu de dezvoltare prietenos și eficient.

Am exersat execuția de instrucțiuni simple în interpretor, învățând cum să utilizăm comenzi de tip print pentru a afișa mesaje și rezultate în consolă. De asemenea, am creat programe simple pentru a exersa concepte precum variabile, expresii și funcții. Am învățat să definim variabile, să efectuăm operații aritmetice și să folosim funcții pentru a organiza și reutiliza codul.

În ceea ce privește instrucțiunile logice și decizionale, am înțeles cum să utilizăm declarații if-else pentru a crea ramificații în cod în funcție de condiții. De asemenea, am explorat instrucțiuni repetitive, cum ar fi buclele for și while, pentru a automatiza execuția repetată a codului.

Pe parcursul acestei lucrări de laborator, am dobândit o bază solidă în limbajul Python și am învățat cum să scriem programe simple, să luăm decizii logice și să cream bucle pentru a gestiona fluxul programelor noastre. Aceste abilități sunt fundamentale pentru dezvoltarea ulterioară în programare și pentru abordarea problemelor mai complexe în Python.

# 4.WEBOGRAFIE

1. Python 3.12.0 documentation disponibil 14.10.2023

( <https://docs.python.org/3/> )

Community for developers disponibil 14.10.2023

1. <https://stackoverflow.com/>

Python 3.12.0 documentation for math library disponibil 14.10.2023

1. <https://docs.python.org/3/library/math.html>