

## MEMORIA PRÁCTICA 2 DSD

ALBA GUISADO FARNES

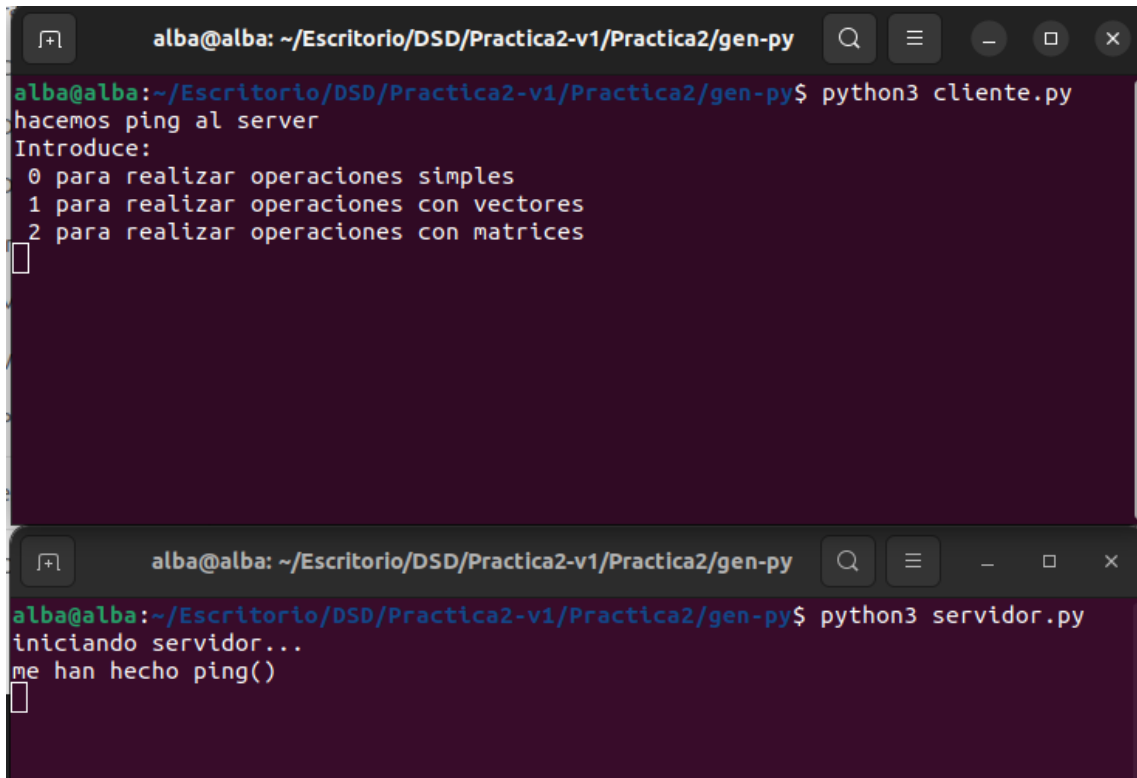
He realizado una calculadora con tres funcionalidades:

- Una calculadora con operaciones algo más sencillas
- Una calculadora de vectores
- Una calculadora de matrices

La estructura usada en nuestra calculadora.thrift es la siguiente:

```
service Calculadora{
    void ping(),
    double suma(1:double num1, 2:double num2),
    double resta(1:double num1, 2:double num2),
    double multiplicacion(1:double num1, 2:double num2),
    double division(1:double num1, 2:double num2),
    double modulo(1:double num1, 2:double num2),
    double elevado(1:double num1, 2:double num2),
    double porcentaje(1:double num1, 2:double num2),
    double raiz(1:double num1, 2:double num2),
    list<double> sumaVector(1:list<double> num1, 2:list<double> num2),
    list<double> restaVector(1:list<double> num1, 2:list<double> num2),
    list<double> productoEscalarV(1:list<double> num1, 2:list<double> num2),
    list<double> productoVectorialV(1:list<double> num1, 2:list<double> num2),
    list<list<double>> sumaMatriz(1:list<list<double>> num1, 2:list<list<double>> num2),
    list<list<double>> restaMatriz(1:list<list<double>> num1, 2:list<list<double>> num2),
    list<list<double>> productoEscalarM(1:list<list<double>> num1, 2:list<list<double>> num2),
    list<list<double>> productoM(1:list<list<double>> num1, 2:list<list<double>> num2),
}
```

Al comenzar nuestro programa preguntará que tipo de calculadora quiere usar:



The image shows two terminal windows. The top window shows a user running 'python3 cliente.py' which prompts them to 'Introduce:' a number. The bottom window shows a user running 'python3 servidor.py' which outputs 'iniciando servidor...' and 'me han hecho ping()'.

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices

```

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()

```

Si introducimos un número inválido la calculadora mostrará que hemos seleccionado un tipo incorrecto.

## 1. Calculadora simple

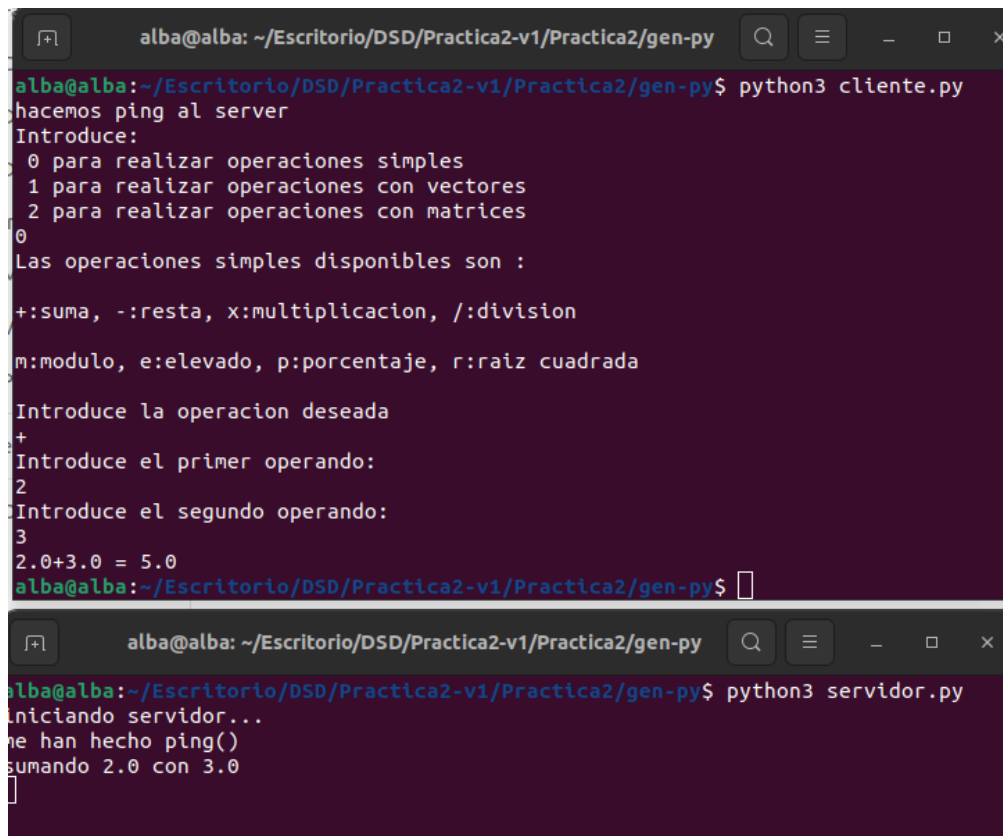
Las operaciones simples implementadas son las siguientes:

- +:suma
- :resta
- x:multiplica
- /:divide
- m:modulo (ej: 4 % 2)
- e:eleve(ej: 2 ^ 4)
- p:calcula porcentajes (ej: 20 p 34, calcula el 20% de 24)
- r: calcula la raíz cuadrada

El primer operando es el número deseado para hacer la raíz  
y el segundo operando el número de precisión  
ej: 4 r 100

Primeramente debemos elegir el tipo de calculadora que queremos usar, después indicar la operación deseada de las disponibles y a continuación introducir el primer operando y después el segundo. Podemos verlo mejor a continuación:

Realizamos una suma:



```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
+
Introduce el primer operando:
2
Introduce el segundo operando:
3
2.0+3.0 = 5.0
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
se han hecho ping()
sumando 2.0 con 3.0
```

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
-
Introduce el primer operando:
4
Introduce el segundo operando:
1
4.0-1.0 = 3.0
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
sumando 2.0 con 3.0
me han hecho ping()
restando 4.0 con 1.0
```

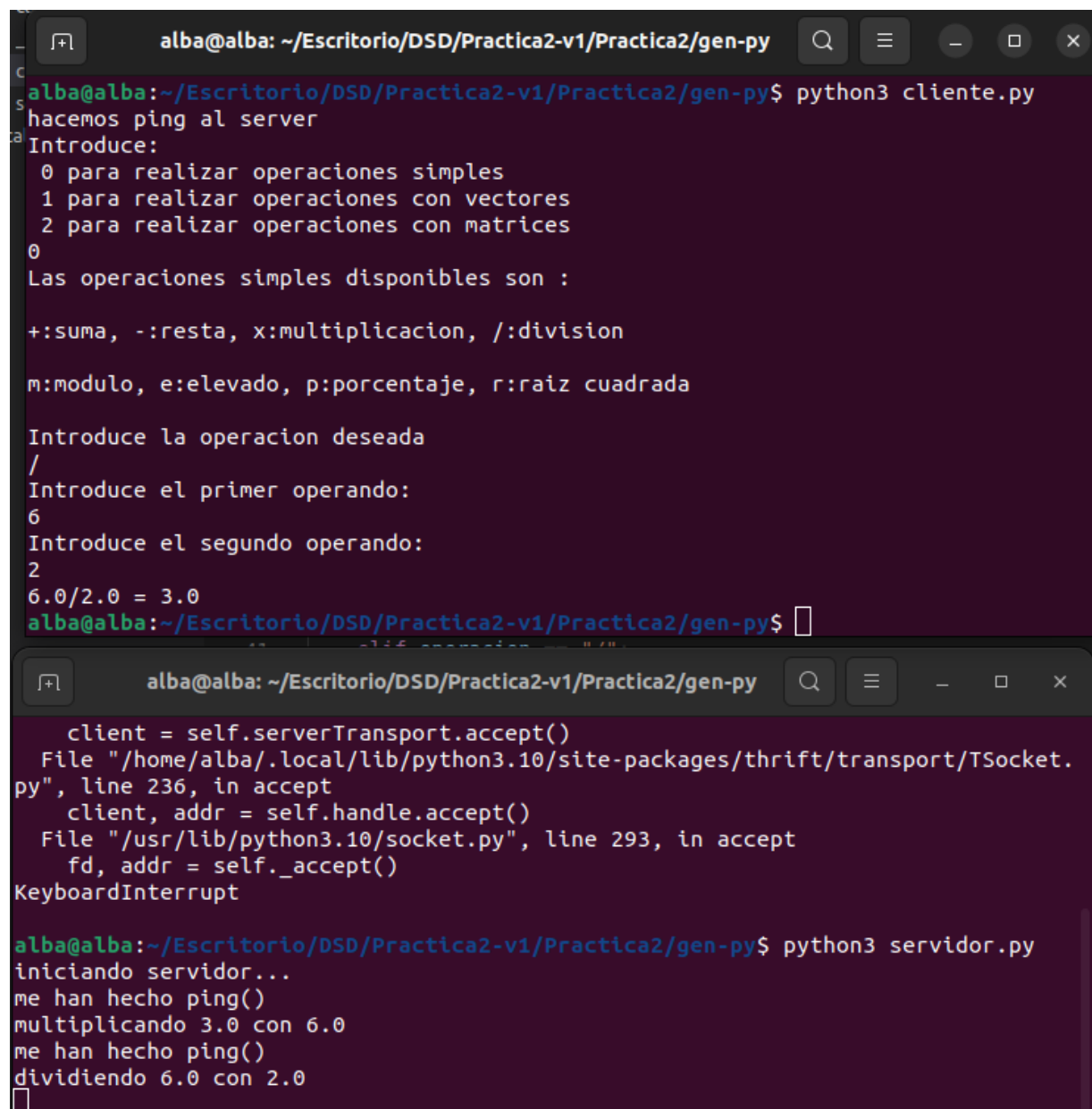
Realizamos una multiplicación:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
x
Introduce el primer operando:
3
Introduce el segundo operando:
6
3.0x6.0 = 18.0
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

File "/home/alba/.local/lib/python3.10/site-packages/thrift/server/TServer.py"
, line 83, in serve
    client = self.serverTransport.accept()
File "/home/alba/.local/lib/python3.10/site-packages/thrift/transport/TSocket.
py", line 236, in accept
    client, addr = self.handle.accept()
File "/usr/lib/python3.10/socket.py", line 293, in accept
    fd, addr = self._accept()
KeyboardInterrupt

alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
multiplicando 3.0 con 6.0
```

Realizamos una división:



```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
/
Introduce el primer operando:
6
Introduce el segundo operando:
2
6.0/2.0 = 3.0
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

client = self.serverTransport.accept()
File "/home/alba/.local/lib/python3.10/site-packages/thrift/transport/TSocket.py", line 236, in accept
    client, addr = self.handle.accept()
File "/usr/lib/python3.10/socket.py", line 293, in accept
    fd, addr = self._accept()
KeyboardInterrupt

alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
multiplicando 3.0 con 6.0
me han hecho ping()
dividiendo 6.0 con 2.0
```

He tenido en cuenta que no se puede dividir entre 0.

También sería un error introducir otro tipo de operación de las que hay permitidas.

Realizamos un ejemplo del módulo:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
m
Introduce el primer operando:
6
Introduce el segundo operando:
4
6.0m4.0 = 2.0
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

py", line 236, in accept
    client, addr = self.handle.accept()
  File "/usr/lib/python3.10/socket.py", line 293, in accept
    fd, addr = self._accept()
KeyboardInterrupt

alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
multiplicando 3.0 con 6.0
me han hecho ping()
dividiendo 6.0 con 2.0
me han hecho ping()
6.0 modulo 4.0
```

Realizamos 2^4:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
e
Introduce el primer operando:
2
Introduce el segundo operando:
4
2.0e4.0 = 24.0
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
2.0 elevado 4.0

```

Realizamos el 20% de 50:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
p
Introduce el primer operando:
20
Introduce el segundo operando:
50
20.0p50.0 = 10.0
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
2.0 elevado 4.0
me han hecho ping()
20.0 porcentaje de 50.0

```

Realizamos la raíz cuadrada de 100 con precisión 10:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
0
Las operaciones simples disponibles son :
h:+:suma, -:resta, x:multiplicacion, /:division
m:modulo, e:elevado, p:porcentaje, r:raiz cuadrada
Introduce la operacion deseada
r
Introduce el primer operando:
100
Introduce el segundo operando:
10
100.0r10.0 = 10.812053925455988
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
2.0 elevado 4.0
me han hecho ping()
20.0 porcentaje de 50.0
me han hecho ping()
10.0 raiz cuadrada con precision 100.0
me han hecho ping()
100.0 raiz cuadrada con precision 10.0

```



## 2. Calculadora de vectores

Las operaciones con vectores realizadas son las siguientes:

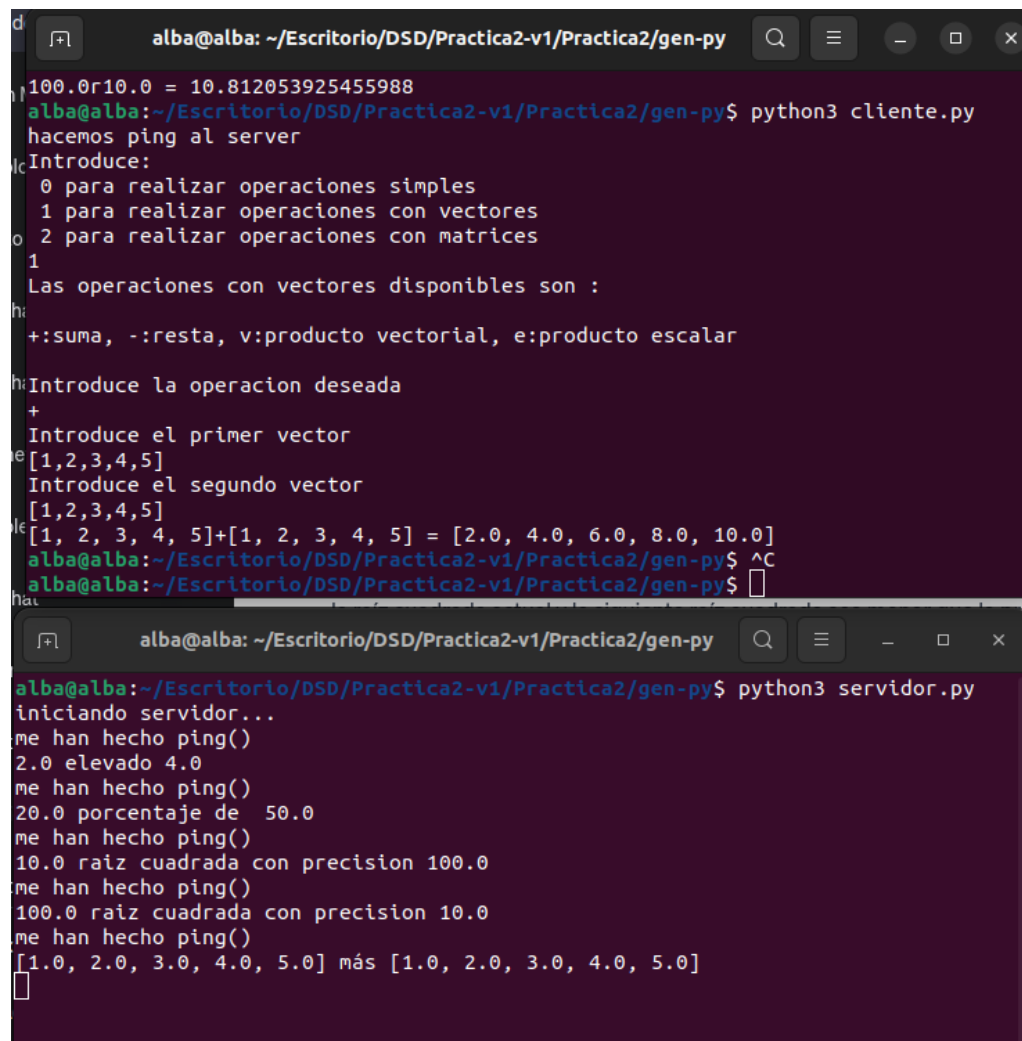
- +:suma de vectores
- :resta de vectores
- x:producto escalar
- v:producto vectorial

Al igual que en la calculadora simple, si introducimos otro tipo de operación nos avisa del error. También comprobamos que los vectores sean del mismo tamaño para poder operar.

**Importante:** Debemos de introducir los vectores de la siguiente forma : [0,1,2,3]. Es decir, entre corchetes y separados por una coma.

A continuación realizaré un ejemplo de cada operación para que entienda su funcionamiento.

Realizamos una suma de vectores:



```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
100.0r10.0 = 10.812053925455988
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
1
Las operaciones con vectores disponibles son :
+:suma, -:resta, v:producto vectorial, e:producto escalar
Introduce la operacion deseada
+
Introduce el primer vector
[1,2,3,4,5]
Introduce el segundo vector
[1,2,3,4,5]
[1, 2, 3, 4, 5]+[1, 2, 3, 4, 5] = [2.0, 4.0, 6.0, 8.0, 10.0]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ ^C
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
2.0 elevado 4.0
me han hecho ping()
20.0 porcentaje de 50.0
me han hecho ping()
10.0 raiz cuadrada con precision 100.0
me han hecho ping()
100.0 raiz cuadrada con precision 10.0
me han hecho ping()
[1.0, 2.0, 3.0, 4.0, 5.0] más [1.0, 2.0, 3.0, 4.0, 5.0]

```

Realizamos resta de vectores:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
^
SyntaxError: invalid syntax
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
  0 para realizar operaciones simples
  1 para realizar operaciones con vectores
  2 para realizar operaciones con matrices
1
h: Las operaciones con vectores disponibles son :
+:suma, -:resta, v:producto vectorial, e:producto escalar
h: Introduce la operacion deseada
-
e: Introduce el primer vector
[1,2]
Introduce el segundo vector
[1,1]
[1, 2]-[1, 1] = [0.0, 1.0]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$
```

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
iniciando servidor...
me han hecho ping()
2.0 elevado 4.0
me han hecho ping()
20.0 porcentaje de 50.0
me han hecho ping()
10.0 raiz cuadrada con precision 100.0
me han hecho ping()
100.0 raiz cuadrada con precision 10.0
me han hecho ping()
[1.0, 2.0, 3.0, 4.0, 5.0] más [1.0, 2.0, 3.0, 4.0, 5.0]
me han hecho ping()
me han hecho ping()
[1.0, 2.0] menos [1.0, 1.0]
```

Realizamos el producto vectorial de dos vectores:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
[1,1]
[1, 2]-[1, 1] = [0.0, 1.0]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
1
h:Las operaciones con vectores disponibles son :
+:suma, -:resta, v:producto vectorial, e:producto escalar
h:
Introduce la operacion deseada
v
e
Introduce el primer vector
[1,2,3]
Introduce el segundo vector
[1,2,3]
[1, 2, 3]v[1, 2, 3] = [0.0, 0.0, 0.0]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$
```

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
2.0 elevado 4.0
me han hecho ping()
20.0 porcentaje de 50.0
me han hecho ping()
10.0 raiz cuadrada con precision 100.0
me han hecho ping()
100.0 raiz cuadrada con precision 10.0
me han hecho ping()
[1.0, 2.0, 3.0, 4.0, 5.0] más [1.0, 2.0, 3.0, 4.0, 5.0]
me han hecho ping()
me han hecho ping()
[1.0, 2.0] menos [1.0, 1.0]
me han hecho ping()
[1.0, 2.0, 3.0] producto vectorial [1.0, 2.0, 3.0]
```

Realizamos el producto escalar de dos vectores:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
[2,3,4,5,6,7]
[1, 1, 1, 1, 1, 1]e[2, 3, 4, 5, 6, 7] = [-1.0, -2.0, -3.0, -4.0, -5.0, -6.0]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
 0 para realizar operaciones simples
 1 para realizar operaciones con vectores
 2 para realizar operaciones con matrices
1
Las operaciones con vectores disponibles son :

+:suma, -:resta, v:producto vectorial, e:producto escalar

Introduce la operacion deseada
e
Introduce el primer vector
[1,2,3,4,5,6]
Introduce el segundo vector
[7,8,9,1,2,3]
[1, 2, 3, 4, 5, 6]e[7, 8, 9, 1, 2, 3] = [7.0, 16.0, 27.0, 4.0, 10.0, 18.0]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
iniciando servidor...
me han hecho ping()
[1.0, 2.0, 3.0, 4.0, 5.0, 6.0] producto escalar [7.0, 8.0, 9.0, 1.0, 2.0, 3.0]
```

### 3. Calculadora de matrices

Las operaciones con matrices realizadas son las siguientes:

- +:suma de matrices
- :resta de matrices
- x:producto de matrices

Al igual que en la calculadora simple, si introducimos otro tipo de operación nos avisa del error. También comprobamos que las matrices sean del mismo tamaño para poder operar en el caso de la suma y la resta. Para el producto de matrices comprobamos que el número de columnas de la primera matriz debe ser igual al número de filas de la segunda matriz.

**Importante:** Primero nos pedirá el número de filas de la primera matriz, debemos de introducir las filas de la siguiente forma : [0,1,2,3]. Es decir, entre corchetes y separados por una coma. Con la segunda matriz sería igual.

A continuación realizaré un ejemplo de cada operación para que entienda su funcionamiento.

Realizamos una suma de matrices:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
2
Las operaciones con matrices disponibles son :
+:suma, -:resta, p:producto de matrices

Introduce la operacion deseada
+
Ingresa el número de filas de la primera matriz: 2

Introduce la primera matriz

Ingresa los elementos de la fila 1:
[1,1]
Ingresa los elementos de la fila 2:
[2,2]
Ingresa el número de filas de la segunda matriz: 2

Introduce el segunda matriz

Ingresa los elementos de la fila 1:
[3,3]
Ingresa los elementos de la fila 2:
[4,4]
[[1, 1], [2, 2]]+[[3, 3], [4, 4]] = [[4.0, 4.0], [6.0, 6.0]]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$
```

Realizamos resta de matrices:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
2 para realizar operaciones con matrices
2
Las operaciones con matrices disponibles son :
+:suma, -:resta, p:producto de matrices

Introduce la operacion deseada
-
Ingresa el número de filas de la primera matriz: 3

Introduce la primera matriz

Ingresa los elementos de la fila 1:
[1,1]
Ingresa los elementos de la fila 2:
[3,3]
Ingresa los elementos de la fila 3:
[5,5]
Ingresa el número de filas de la segunda matriz: 3

Introduce la segunda matriz

Ingresa los elementos de la fila 1:
[2,2]
Ingresa los elementos de la fila 2:
[4,4]
Ingresa los elementos de la fila 3:
[6,6]
[[1, 1], [3, 3], [5, 5]]-[[2, 2], [4, 4], [6, 6]] = [[-1.0, -1.0], [-1.0, -1.0],
[-1.0, -1.0]]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$
```

Realizamos el producto de matrices:

```
alba@alba: ~/Escritorio/DSD/Practica2-v1/Practica2/gen-py
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 cliente.py
hacemos ping al server
Introduce:
0 para realizar operaciones simples
1 para realizar operaciones con vectores
2 para realizar operaciones con matrices
2
Las operaciones con matrices disponibles son :
+:suma, -:resta, p:producto de matrices
Introduce la operacion deseada
p
Ingrese el número de filas de la primera matriz: 2
Introduce la primera matriz
Ingrese los elementos de la fila 1:
[1,2]
Ingrese los elementos de la fila 2:
[3,4]
Ingrese el número de filas de la segunda matriz: 2
Introduce el segunda matriz
Ingrese los elementos de la fila 1:
[5,6]
Ingrese los elementos de la fila 2:
[7,8]
[[1, 2], [3, 4]]p[[5, 6], [7, 8]] = [[19.0, 22.0], [43.0, 50.0]]
alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$

alba@alba:~/Escritorio/DSD/Practica2-v1/Practica2/gen-py$ python3 servidor.py
Iniciando servidor...
Se han hecho ping()
[[1.0, 2.0], [3.0, 4.0]] producto [[5.0, 6.0], [7.0, 8.0]]
```