

# Práctica 2

Llamada a procedimiento remoto (RPC)

## Parte 2 - Apache Thrift

Luis Miguel Guirado Bautista  
Curso 2024-2025  
Universidad de Granada

---

<b>Introducción</b>	<b>2</b>
<b>Representación de datos compartidos</b>	<b>2</b>
<b>Implementación y funcionamiento</b>	<b>3</b>
Parte troncal. Operaciones básicas, factoriales, MCM y MCD	3
Operaciones basicas	3
Factoriales	3
Minimo Comun Multiplo	4
Maximo Comun Divisor	4
Parte adicional. Operaciones compuestas y vectoriales.	5
Operaciones compuestas	5
Operaciones vectoriales	6
Servidor	7
Cliente	7

## Introducción

Este documento contiene información acerca del desarrollo de la calculadora que se ha pedido para esta práctica. Se hablará sobre la definición y representación de datos y estructuras de datos compartidas, el funcionamiento del programa cliente y servidor, la comunicación entre estos y las salidas que muestran. Entre las operaciones implementadas se encuentran:

- Operaciones básicas
- Factoriales
- Mínimo Común Múltiplo
- Máximo Común Divisor
- Operaciones compuestas
- Operaciones vectoriales
- Vector por escalar

## Representación de datos compartidos

Se han definido los siguientes tipos:

- Número como `double`
  - Usado como operandos en la mayoría de operaciones
- Operador como `TipoOperador` (definido más adelante)
  - Usado en las operaciones básicas, compuestas y vectoriales
- `ListaEnteros` como lista de enteros de 64b
  - Usado en las operaciones de MCM y MCD
- `Numeros` como lista de `Numero`
  - Usado en las operaciones compuestas para guardar los operandos
- `Operadores` como lista de `Operador`
  - Usado en las operaciones compuestas para guardar los operadores
- `Vector` como lista de `Numero`
  - Usado en operaciones vectoriales

Se ha definido la enumeración `TipoOperador`:

- `SUMA = 1`
- `RESTA = 2`
- `MULTIPLICACION = 3`
- `DIVISION = 4`

Se han definido las siguientes estructuras de datos

- `Operacion`. Guarda los datos de una operación básica: sus operandos y su operador
- `OperacionCompuesta`. Guarda los datos de una operación compuesta: sus operandos y operadores
- `OperacionVectorial`. Guarda los datos de una operación vectorial: sus dos vectores operandos y su operador

También se ha definido la interfaz del servidor, que corresponde con la lista de operaciones implementadas

## Implementación y funcionamiento

Se ha realizado de manera muy similar a la version de RPC Sun.

El programa cliente interactua con un menú de opciones y entrada guiada por el programa para poder enviar las solicitudes con sus datos correspondientes al servidor, previamente procesados a una estructura de datos como las definidas anteriormente.

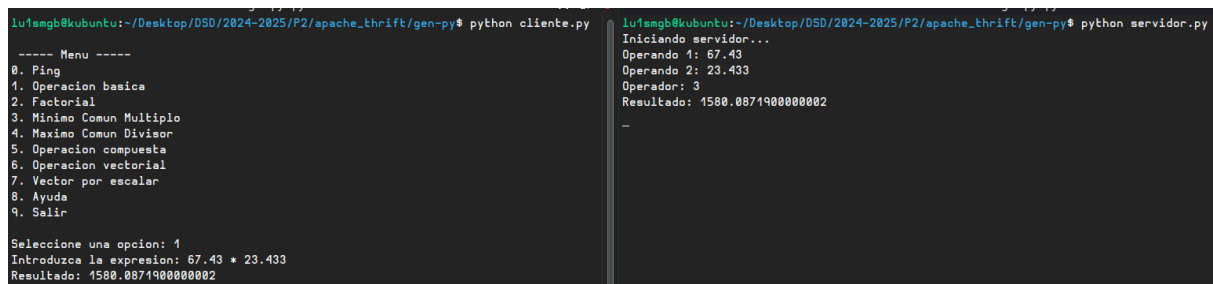
El servidor se encargará de procesar las solicitudes recibidas en el resultado de la operación y enviarlas al cliente que envió la solicitud.

En resumen, el cliente se encarga de convertir la entrada del usuario a una estructura de datos y el servidor se encarga de los cálculos.

## Parte troncal. Operaciones básicas, factoriales, MCM y MCD

### Operaciones basicas

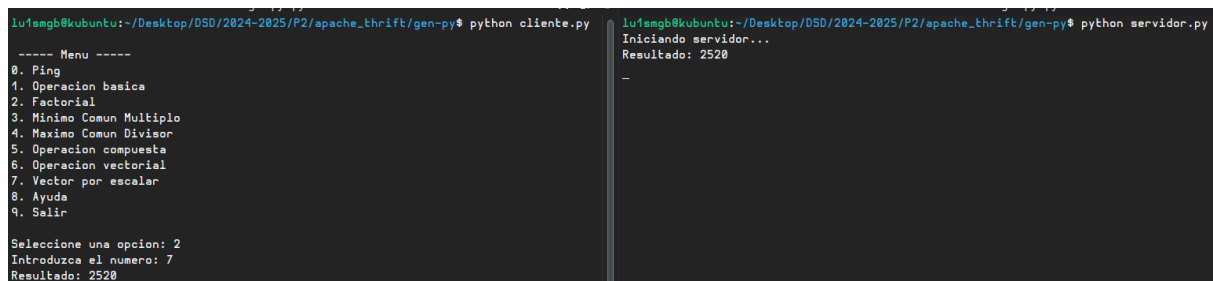
El cliente recogerá los operandos enteros o decimales y el operador como carácter proporcionados por el usuario y lo empaqueta en una Operacion para enviarlo al servidor. Despues el servidor realiza la operación aritmética con los operandos en función del operador y devuelve el resultado al cliente.



The screenshot shows two terminal windows. The left window is the client, and the right window is the server. The client displays a menu with options 0-9. Option 1, 'Operacion basica', is selected. The user enters the expression '67.43 \* 23.433'. The client sends this to the server. The server window shows 'Iniciando servidor...', followed by 'Operando 1: 67.43', 'Operando 2: 23.433', 'Operador: 3', and finally 'Resultado: 1580.0871900000002'.

### Factoriales

El cliente preguntará por el número del cual hay que calcular el factorial al cliente y lo enviará al servidor para que se encargue del trabajo matemático y envíe el resultado de vuelta al cliente. El factorial se calcula de forma iterativa y sirve tanto para numeros positivos como negativos.

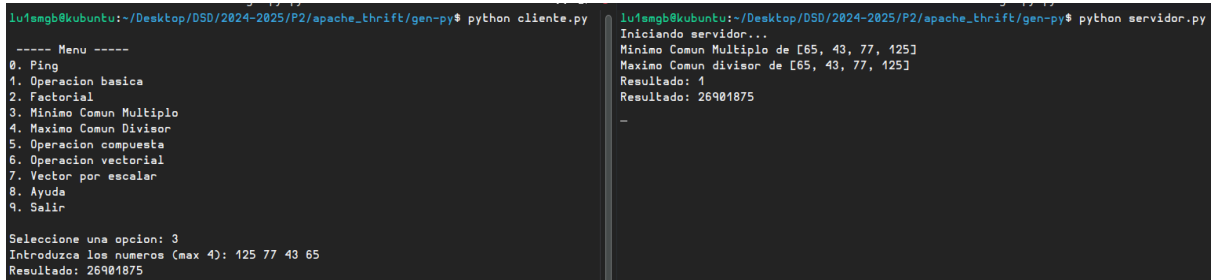


The screenshot shows two terminal windows. The left window is the client, and the right window is the server. The client displays the same menu as before. Option 2, 'Factorial', is selected. The user enters the number '7'. The client sends this to the server. The server window shows 'Iniciando servidor...' followed by 'Resultado: 2520'.

## Minimo Comun Multiplo

El cliente pregunta por los numeros y se queda con los cuatro primeros, los empaqueta en una lista de enteros y los envía al servidor. El servidor filtra los repetidos y calcula el MCM dividiendo el valor absoluto de los cuatro numeros entre el MCD de estos.

$$MCM(a, b, c, d) = \frac{|a \times b \times c \times d|}{MCD(a, b, c, d)}$$



```

luis@luis@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py
----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

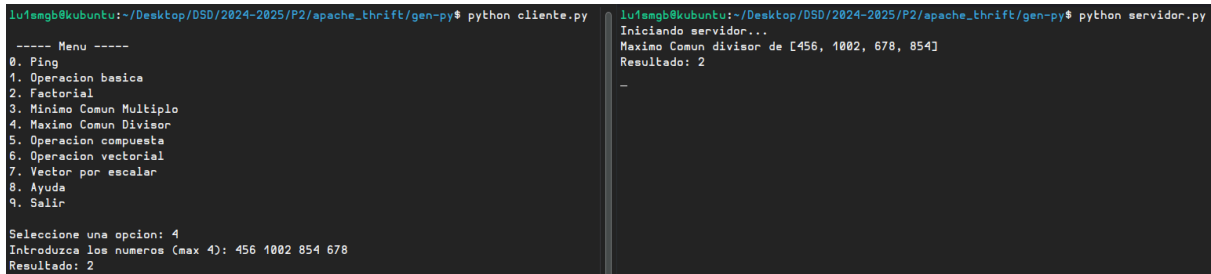
Seleccione una opcion: 3
Introduzca los numeros (max 4): 125 77 43 65
Resultado: 26901875

luis@luis@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python servidor.py
Iniciando servidor...
Minimo Comun Multiplo de [65, 43, 77, 125]
Maximo Comun divisor de [65, 43, 77, 125]
Resultado: 1
Resultado: 26901875

```

## Maximo Comun Divisor

El cliente pregunta por los numeros y se queda con los cuatro primeros, los empaqueta y los envía al servidor. El servidor filtra los repetidos y calcula el máximo comun divisor de manera iterativa tomando como valor candidato desde el mínimo de los valores proporcionados hasta el 1. Si el candidato divide a todos los valores, se envía el valor al cliente.



```

luis@luis@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py
----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

Seleccione una opcion: 4
Introduzca los numeros (max 4): 456 1002 854 678
Resultado: 2

luis@luis@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python servidor.py
Iniciando servidor...
Maximo Comun divisor de [456, 1002, 678, 854]
Resultado: 2

```

## Parte adicional. Operaciones compuestas y vectoriales.

### Operaciones compuestas

El cliente lee una expresión compuesta y la procesa en una estructura de datos OperacionCompuesta, separando los operadores y operandos en listas dedicadas. Una vez las envía al servidor, resolverá la expresión reduciéndose hasta un solo operando mediante el cálculo de suboperaciones, escogiendo el operador más prioritario y sus operandos adyacentes y actualizando las listas de operadores y operandos con cada suboperación.

```

luismgb@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py
----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

Seleccione una opcion: 5
Introduzca la expresion: 1 + 2 - 3 * 4 / 5
[1.0, 2.0, 3.0, 4.0, 5.0]
[1, 2, 3, 4]
Resultado: 0.6000000000000001

----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

Seleccione una opcion: _

luismgb@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python servidor.py
Iniciando servidor...
Buscando operador MULTIPLICACION
Operador MULTIPLICACION encontrado en indice 2
3.0 4.0 MULTIPLICACION
Operando 1: 3.0
Operando 2: 4.0
Operador: 3
Resultado: 12.0
[1.0, 2.0, 3.0, 4.0, 5.0]
['SUMA', 'RESTA', 'MULTIPLICACION', 'DIVISION']
2
[1.0, 2.0, 12.0, 5.0]
['SUMA', 'RESTA', 'DIVISION']
Buscando operador MULTIPLICACION
Buscando operador DIVISION
Operador DIVISION encontrado en indice 2
12.0 5.0 DIVISION
Operando 1: 12.0
Operando 2: 5.0
Operador: 4
Resultado: 2.4
[1.0, 2.0, 12.0, 5.0]
['SUMA', 'RESTA', 'DIVISION']
2
[1.0, 2.0, 2.4]
['SUMA', 'RESTA']
Buscando operador MULTIPLICACION
Buscando operador DIVISION
Buscando operador SUMA
Operador SUMA encontrado en indice 0
1.0 2.0 SUMA
Operando 1: 1.0
Operando 2: 2.0
Operador: 1
Resultado: 3.0
[1.0, 2.0, 2.4]
['SUMA', 'RESTA']
0
[3.0, 2.4]
['RESTA']
Buscando operador MULTIPLICACION
Buscando operador DIVISION
Buscando operador SUMA
Buscando operador RESTA
Operador RESTA encontrado en indice 0
3.0 2.4 RESTA
Operando 1: 3.0
Operando 2: 2.4
Operador: 2
Resultado: 0.6000000000000001
[3.0, 2.4]
['RESTA']
0
[0.6000000000000001]
[]

```

## Operaciones vectoriales

El cliente lee un conjunto de numeros separados por espacios para preparar el primer vector, despues lee el operando y el segundo vector tal y como lo haria con el primero.

Empaqueta los datos en una OperacionVectorial y la envía la operacion al servidor para que se encargue de procesar el vector resultante.

También se puede realizar la multiplicación de un vector por un escalar, en el que el cliente solo lee un vector y un entero. Además, si en el caso anterior, si con el operador de multiplicacion, el segundo vector solo tiene un valor, se considerará una operación de vector por escalar y se procesará como tal.

```
luis@ubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py
----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

Seleccione una opcion: 6
Introduzca los numeros del vector 1: 1 2 3 4 5
Introduzca el operador: *
Introduzca los numeros del vector 2: 2 4 6 8 10
Resultado: [2.0, 8.0, 18.0, 32.0, 50.0]
```

```
luis@ubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python servidor.py
Iniciando servidor...
Resultado: [2.0, 8.0, 18.0, 32.0, 50.0]
```

```
luis@ubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py
----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

Seleccione una opcion: 7
Introduzca los numeros del vector 1: 1 2 3 4 5
Introduzca el escalar: 9
Resultado: [9.0, 18.0, 27.0, 36.0, 45.0]
```

```
luis@ubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python servidor.py
Iniciando servidor...
Resultado: [9.0, 18.0, 27.0, 36.0, 45.0]
```

```
luis@ubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py
----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

Seleccione una opcion: 6
Introduzca los numeros del vector 1: 1 2 3 4 5
Introduzca el operador: *
Introduzca los numeros del vector 2: 9
Resultado: [9.0, 18.0, 27.0, 36.0, 45.0]
```

```
luis@ubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python servidor.py
Iniciando servidor...
Resultado: [9.0, 18.0, 27.0, 36.0, 45.0]
```

## Servidor

Para abrir el servidor, se ejecutará en el directorio gen-py el siguiente comando:

```
python servidor.py
```

Ahora solo queda ejecutar el cliente para poder enviarle operaciones

## Cliente

Para abrir el cliente, se ejecutará en el directorio gen-py el siguiente comando:

```
python cliente.py
```

Se nos mostrará un menú con las diferentes funcionalidades que podemos invocar

```
lu1smgb@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py

----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir
```

Al escoger la opcion 8, se nos mostrara la siguiente información de ayuda

```
----- Ayuda -----
Los operandos pueden ser numeros enteros o decimales
Los operadores pueden ser los siguientes caracteres: +, -, * o /
Imprescindible separar los operadores y los operandos con espacios en operaciones compuestas
En las operaciones vectoriales no se puede utilizar / como operador
Prioridad de los operandos en las operaciones compuestas (descendente): * -> / -> + y -
Los operadores a la izquierda tienen mas prioridad
-----
```

Al escoger la opcion 9 (o Ctrl-C), saldremos de la calculadora

```
Seleccione una opcion: ^C
Adios! :)
lu1smgb@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ python cliente.py

----- Menu -----
0. Ping
1. Operacion basica
2. Factorial
3. Minimo Comun Multiplo
4. Maximo Comun Divisor
5. Operacion compuesta
6. Operacion vectorial
7. Vector por escalar
8. Ayuda
9. Salir

Seleccione una opcion: 9
Adios! :)
lu1smgb@kubuntu:~/Desktop/DSD/2024-2025/P2/apache_thrift/gen-py$ _
```