

# DSD P1

Raúl Castro Moreno

Marzo 2021

## 1 Descripción

Para esta práctica he realizado 2 tipos de calculadoras, una básica y otra de matrices, todo esto usando la tecnología de RPC Sun.

## 2 Calculadora Básica

En esta calculadora tenemos la opción de realizar las 4 operacion elementales, pudiendo usar números decimales (float). Las operaciones son:

- Suma
- Resta
- Multiplicación
- División

Sobre esta calculadora se ha aplicado también una gestión de errores la cuál permite tratar fallos como el de realizar una división entre 0.

### 2.1 Entrada

Para la entrada de esta calculadora tendremos que usar el siguiente formato de entrada de argumentos:

```
./cliente (host) (operando1) (signo operacion) (operando2)
```

Un ejemplo sería:

```
./cliente localhost 5 + 7
```

### 2.2 Generación

Para generar los ficheros con la tecnología de RPC Sun, hemos usado el archivo calculadora.x (FIGURA 1)

### 2.3 Ejemplo de ejecución

Aqui muestro un ejemplo de ejecución donde pruebo todas las operaciones y la que da error. (Los signos \* y / deben llevar delante la barra invertida porque son caracteres reservados de la shell) (FIGURA 2)

```

union calculadora_response switch(int err) {
    case 0:
        float result;
    default:
        void;
};

program CALCULADORA {
    version CALCV1 {
        calculadora_response SUMAR (float,float) = 1;
        calculadora_response RESTAR (float,float) = 2;
        calculadora_response MULTIPLICAR (float,float) = 3;
        calculadora_response DIVIDIR (float,float) = 4;
    } = 1;
} = 0x20000001;

```

Figure 1: calculadora.x

```

ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/p1/basica$ ./cliente localhost 5 + 7
Resultado de la operación 5.000000 + 7.000000 = 12.000000
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/p1/basica$ ./cliente localhost 5 - 7
Resultado de la operación 5.000000 - 7.000000 = -2.000000
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/p1/basica$ ./cliente localhost 5 \* 7
Resultado de la operación 5.000000 * 7.000000 = 35.000000
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/p1/basica$ ./cliente localhost 5 \/ 7
Resultado de la operación 5.000000 / 7.000000 = 0.714286
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/p1/basica$ ./cliente localhost 5 \/ 0
Ha habido algún error en la operación: Operation not permitted
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/p1/basica$ 

```

Figure 2: Ejemplo de ejecución de calculadora básica

### 3 Calculadora de Matrices

En esta calculadora podemos realizar con matrices las siguientes operaciones, pudiendo usar también números decimales (float) en los valores que guardan las matrices:

- Suma
- Resta
- Multiplicación

Aquí tratamos también posibles errores como puede ser que se introduzcan 2 matrices las cuales para realizar la multiplicación, no tengan las dimensiones necesarias (Número de columnas de la primera matriz = Número de filas de la segunda matriz)

#### 3.1 Entrada

Para la entrada de esta calculadora tendremos que usar el siguiente formato de entrada de argumentos siendo la primera la matriz A y la segunda la matriz B:

`./cliente (host) (signo operacion) (filasA) (columnasA) (valorA1) (valorA2) ... (filasB) (columnasB) (valorB1) (valorB2) ...`

Un ejemplo sería:

`./cliente localhost + 2 2 1 2 3 4 2 2 5 6 7 8`

Esto sería sumar una matriz 2\*2 con valores 1 2 3 4 con otra matriz 2\*2 con valores 5 6 7 8

#### 3.2 Generación

Para generar los ficheros con la tecnología de RPC Sun, hemos usado el fichero calculadora-matriz.x (FIGURA 3)

```

struct matriz{
    u_int fil;
    u_int col;
    float mat<->;
};

union matriz_response switch(int err) {
    case 0:
        matriz result;
    default:
        void;
};

program CALCULADORA MAT {
    version CALC_MAT.V1 0;
    matriz_response SUMA_MATRICES(matriz,matriz) = 1;
    matriz_response RESTA_MATRICES(matriz,matriz) = 2;
    matriz_response MULT_MATRICES(matriz,matriz) = 3;
    0 = 1;
} = 0x2200001;

```

Figure 3: calculadora.x

### 3.3 Ejemplo de ejecución

Aquí muestro un ejemplo de ejecución donde pruebo todas las operaciones y las que dan error. (El signo \* debe llevar delante la barra invertida porque es un carácter reservado de la shell) (FIGURA 4)

```

ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/pl/matrices$ ./cliente localhost + 3 3 1 2 3 4 5 6 7 8 9 3 3 1 2 3 4 5 6 7 8 9
Imprimiendo el resultado
2.000000 4.000000 6.000000
8.000000 10.000000 12.000000
14.000000 16.000000 18.000000
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/pl/matrices$ ./cliente localhost + 3 3 1 2 3 4 5 6 7 8 9 3 2 1 2 3 4 5 6
Ha habido algún error en la operación: Operation not permitted
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/pl/matrices$ ./cliente localhost - 3 3 1 2 3 4 5 6 7 8 9 3 3 1 2 3 4 5 6 7 8 9
Imprimiendo el resultado
0.000000 0.000000 0.000000
0.000000 0.000000 0.000000
0.000000 0.000000 0.000000
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/pl/matrices$ ./cliente localhost - 3 3 1 2 3 4 5 6 7 8 9 3 2 1 2 3 4 5 6
Ha habido algún error en la operación: Operation not permitted
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/pl/matrices$ ./cliente localhost \* 3 3 1 2 3 4 5 6 7 8 9 3 3 1 2 3 4 5 6 7 8 9
Imprimiendo el resultado
30.000000 36.000000 42.000000
66.000000 81.000000 96.000000
102.000000 126.000000 150.000000
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/pl/matrices$ ./cliente localhost \* 3 3 1 2 3 4 5 6 7 8 9 2 3 1 2 3 4 5 6
Ha habido algún error en la operación: Operation not permitted
ubuntu@Ubu18:/home/raul/Escritorio/Home/DSD/pl/matrices$

```

Figure 4: Ejemplo de ejecución de calculadora de matrices