

MEMORIA PRÁCTICA 2 DSD

ALBA GUIADO FARNES

He realizado una calculadora con tres funcionalidades:

- Una calculadora con operaciones algo más sencillas
- Una calculadora de vectores con 3 coordenadas
- Una calculadora de matrices 3x3

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

```
typedef float float_type; /*Tipo dato float*/
typedef char *char_type; /*Tipo dato *char*/
typedef struct _vector vector_type; /*Tipo dato vector*/
typedef struct matriz matriz_type; /*Tipo dato matriz*/
typedef struct namecalculo calculo1;
typedef struct namecalculo2 calculo2;
typedef struct namecalculo3 calculo3;
typedef float resultado1; /*Variable del resultado 1*/
typedef struct _vector resultado2;
typedef struct matriz resultado3;

struct namecalculo {
    float_type uno; /*Primer operando*/
    char_type operacion; /* Tipo operacion*/
    float_type dos; /*Segundo operando*/
};

struct _vector{
    float_type f1;
    float_type f2;
    float_type f3;
};

struct namecalculo2{
    vector_type uno;
    char_type operacion;
    vector_type dos;
};

struct matriz{
    vector_type v1;
    vector_type v2;
    vector_type v3;
};

struct namecalculo3{
    matriz_type uno;
    char_type operacion;
    matriz_type dos;
};
```

```

};

/* la siguiente union se utiliza para discriminar entre llamadas con exito y llamadas con errores */
union calculadora_res1 switch (int errno) {
    case 0:
        resultado1 solucion; /* sin error: listado del directorio */
    default:
        void; /* con error: nada */
};

union calculadora_res2 switch (int errno) {
    case 0:
        resultado2 solucion; /* sin error: listado del directorio */
    default:
        void; /* con error: nada */
};

union calculadora_res3 switch (int errno) {
    case 0:
        resultado3 solucion; /* sin error: listado del directorio */
    default:
        void; /* con error: nada */
};

program CALCULADORAPROG {
    version CALCULADORA1{
        calculadora_res1 CALCULADORA(calculo1) = 1;
    } =1;
    version CALCULADORA2{
        calculadora_res2 CALCULADORA(calculo2) = 1;
    } =2;
    version CALCULADORA3{
        calculadora_res3 CALCULADORA(calculo3) = 1;
    } =3;
} = 0x20000155;

```

1. Calculadora simple

La entrada debe de ser de tal forma: <programa> <máquina> <float> <operación> <float>

Las operaciones simples implementadas son las siguientes:

```

+:suma
-:resta
x:multiplica
/:divide
m:modulo (ej: 4 % 2)
^:eleva      (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 presición
    ej: 4 r 100
g: es un conversor de grados a radianes y viceversa
    el segundo operando indica si convertimos a radianes o
grados;
    0 para radianes, 1 para grados
    ej: 4 g 0 -> Convertimos de 4 grados a radianes

```

ej: 4 g 1 -> Convertimos de 4 radianes a grados

```
alba@alba: ~/Escritorio/DSD/calculadora
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 + 4
Resultado: 8.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 - 4
Resultado: 0.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 x 4
Resultado: 16.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 / 4
Resultado: 1.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 m 2
Resultado: 0.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 ^ 2
Resultado: 16.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 20 p 50
Resultado: 10.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 2 r 1000
Resultado: 1.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 r 100
Resultado: 2.000000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 114 r 10
0
Resultado: 29.500000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 / 0
No se puede dividir entre 0
alba@alba:~/Escritorio/DSD/calculadora$
```

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.

```
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 4 w 0
La operación introducida es incorrecta, por favor seleccione una válida.

Para usar la calculadora simple: ./calculadora_client server_host float operación float
Las operaciones simples implementadas son las siguientes:
Pulse + para sumar
Pulse - para restar
Pulse x para multiplicar
Pulse / para dividir
Pulse m para realizar el módulo
Pulse p para calcular porcentajes, calcula porcentajes (ej: 20 p 34, calcula el 20 por ciento de 34)
Pulse r para calcular la raíz de un número, ej: 4 r precisión, calcula la raíz de 4 con una cierta precisión
Pulse ^ para realizar el elevado

alba@alba:~/Escritorio/DSD/calculadora$
```

2. Calculadora de vectores de tres coordenadas

La entrada debe de ser de tal forma: <programa> <máquina> <float> <float> <float>
<operación> <float> <float> <float>

Es decir, ./calculadora_client server_host vector1.x vector1.y vector1.z operación vector2.x vector2.y vector2.z

Las operaciones con vectores de 3 coordenadas realizadas son las siguientes:

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

```
alba@alba:~/Escritorio/DSD/calculadora$ ^C
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 1 2 3 +
1 2 3
Vector resultante: 2.000 4.000 6.000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 1 2 3 -
1 2 3
Vector resultante: 0.000 0.000 0.000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 1 2 3 x
1 2 3
Vector resultante: 1.000 4.000 9.000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 1 2 3 v
1 2 3
Vector resultante: 0.000 0.000 0.000
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 1 2 3 v
3 3 3
Vector resultante: -3.000 6.000 -3.000
alba@alba:~/Escritorio/DSD/calculadora$
```

Al igual que en la calculadora simple, si introducimos otro tipo de operación nos avisa del error:

```
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost 1 2 3 t
3 3 3
La operación introducida es incorrecta, por favor seleccione una válida.

Para usar la calculadora de vectores de 3 coordenadas:./calculadora_client server_host vector1.x vector1.y vector1.z operación vector2.x vector2.y vector2.z
Las operaciones con vectores de 3 coordenadas implementadas son las siguientes:
Pulse + para sumar los vectores
Pulse - para restar los vectores
Pulse x para hacer el producto escalar de los vectores
Pulse v para hacer el producto vectorial de los vectores

Pulse v para hacer el producto escalar de matrices
alba@alba:~/Escritorio/DSD/calculadora$
```

3. Calculadora de matrices 3x3

La entrada debe de ser de tal forma: <programa> <máquina> <operación> <float> <float> <float> <float> <float> <float> <float> <float> <float> <float> <float> <float> <float> <float> <float> <float>

Es decir, ./calculadora_client server_host operacion matriz1[0][0] matriz1[0][1] matriz1[0][2] matriz1[1][0] matriz1[1][1] matriz1[1][2] matriz1[2][0] matriz1[2][1] matriz1[2][2] matriz2[0][0] matriz2[0][1] matriz2[0][2] matriz2[1][0] matriz2[1][1] matriz2[1][2] matriz2[2][0] matriz2[2][1] matriz2[2][2]

Las operaciones con matrices 3x3 realizadas son las siguientes:

- +:suma de matrices
- .resta de matrices
- x:producto de matrices
- e:producto escalar de matrices

```
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost + 1 2 3
4 5 6 7 8 9 1 2 3 4 5 6 7 8 9
Matriz resultante:
2.00 4.00 6.00
8.00 10.00 12.00
14.00 16.00 18.00
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost - 1 2 3
4 5 6 7 8 9 1 2 3 4 5 6 7 8 9
Matriz resultante:
0.00 0.00 0.00
0.00 0.00 0.00
0.00 0.00 0.00
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost x 1 2 3
4 5 6 7 8 9 1 2 3 4 5 6 7 8 9
Matriz resultante:
30.00 36.00 42.00
66.00 81.00 96.00
102.00 126.00 150.00
alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost e 1 2 3
4 5 6 7 8 9 1 2 3 4 5 6 7 8 9
Matriz resultante:
1.00 4.00 9.00
16.00 25.00 36.00
49.00 64.00 81.00
alba@alba:~/Escritorio/DSD/calculadora$
```

Al igual que en la calculadora simple, si introducimos otro tipo de operación nos avisa del error:

```

alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost a 1 2 3
4 5 6 7 8 9 1 2 3 4 5 6 7 8 9
La operación introducida es incorrecta, por favor seleccione una válida.

Para usar la calculadora de matrices 3x3:./calculadora_client server_host operac
ion matriz1[0][0] matriz1[0][1] matriz1[0][2] matriz1[1][0] matriz1[1][1] matriz
1[1][2] matriz1[2][0] matriz1[2][1] matriz1[2][2] matriz2[0][0] matriz2[0][1] ma
triz2[0][2] matriz2[1][0] matriz2[1][1] matriz2[1][2] matriz2[2][0] matriz2[2][1
] matriz2[2][2]
Las operaciones con matrices 3x3 implementadas son las siguientes:
Pulse + para sumar las matrices
Pulse - para restar las matrices
Pulse x para hacer el producto de matrices
Pulse e para hacer el producto escalar de matrices

alba@alba:~/Escritorio/DSD/calculadora$

```

En cualquier caso, si introducimos el número de parámetros incorrecto nos avisaría del uso del programa calculadora (si no es ni 5 que son los utilizados para la calculadora simple, ni 9, que son los utilizados para la calculadora de vectores, ni 21, los utilizados para la calculadora de matrices). Podemos verlo en la siguiente imagen:

```

alba@alba:~/Escritorio/DSD/calculadora$ ./calculadora_client localhost
Para usar la calculadora simple: ./calculadora_client server_host float operación flo
at
Las operaciones simples implementadas son las siguientes:
Pulse + para sumar
Pulse - para restar
Pulse x para multiplicar
Pulse / para dividir
Pulse m para realizar el módulo
Pulse p para calcular porcentajes, calcula porcentajes (ej: 20 p 34, calcula el 20 po
r ciento de 34)
Pulse r para calcular la raíz de un número, ej: 4 r precisión, calcula la raíz de 4 c
on una cierta precisión
Pulse ^ para realizar el elevado

Para usar la calculadora de vectores de 3 coordenadas:./calculadora_client server_hos
t vector1.x vector1.y vector1.z operación vector2.x vector2.y vector2.z
Las operaciones con vectores de 3 coordenadas implementadas son las siguientes:
Pulse + para sumar los vectores
Pulse - para restar los vectores
Pulse x para hacer el producto escalar de los vectores
Pulse v para hacer el producto vectorial de los vectores

Para usar la calculadora de matrices 3x3:./calculadora_client server_host operacion m
atriz1[0][0] matriz1[0][1] matriz1[0][2] matriz1[1][0] matriz1[1][1] matriz1[1][2] ma
triz1[2][0] matriz1[2][1] matriz1[2][2] matriz2[0][0] matriz2[0][1] matriz2[0][2] ma
triz2[1][0] matriz2[1][1] matriz2[1][2] matriz2[2][0] matriz2[2][1] matriz2[2][2]
Las operaciones con matrices 3x3 implementadas son las siguientes:
Pulse + para sumar las matrices
Pulse - para restar las matrices
Pulse x para hacer el producto de matrices
Pulse e para hacer el producto escalar de matrices

alba@alba:~/Escritorio/DSD/calculadora$

```