

1. Utilizando las siguientes estructuras y definiciones para representar expresiones matemáticas simbólicas:

```
typedef enum ExpT {
    EXP_CONSTANT,
    EXP_PLUS,
    EXP_MINUS,
    EXP_MULT,
    EXP_DIV,
    EXP_SIN,
    EXP_COS,
    EXP_TAN,
    EXP_VARIABLE
} ExpType;

typedef struct Exp_str {
    ExpType tipo;
    union {
        double value;           // constants
        struct {
            struct Exp_str *Arg1, *Arg2;    // binarias
        } binary;
        struct {
            struct Exp_str *Arg1;           // unarias
        } unary;
    } args;
} *Exp;
```

Diseñar e implementar las siguientes funciones:

```
// Imprima la expresión en forma legible
void PrintEXP(Exp expr);

// Evalúe la expresión, en el valor de la variable X
double Evaluate(Exp expr, double X);

// Libere todos los recursos asociados a expr.
void Destroy(Exp expr);

// Calcule la expresión derivada analítica de la expresión
Exp Derivate(Exp expr);

// Cree y retorne una expresión binaria
Exp CreateBinaryEXP(ExpType type, Exp expr1, Exp expr2);

// Cree y retorne una expresión unaria
Exp CreateUnaryEXP(ExpType type, Exp expr1);

// Cree y retorne una expresión constante
Exp CreateConstEXP(double value);

// Cree y retorne una expresión de variable
Exp CreateVarEXP();

// Cree y retorne un clon de la expresión recibida
Exp CloneEXP(Exp e);
```

2. Realice un programa que utilice las funciones del ejercicio anterior:

- cree la siguiente expresión: $\frac{\sin(3x+4)}{2+3 \cos(\frac{x}{2+x})}$
- Imprima la expresión del punto anterior
- La evalúe la expresión en el rango $[0, 1]$ con saltos de 0.01
- Obtenga e imprima la expresión derivada.
- Destruya la expresión creada.

Tenga en cuenta que la expresión a representar puede verse como una estructura con la siguiente topología:

