PRÁCTICA Nº 1

En esta prueba práctica se evaluará tu capacidad para diseñar y programar plantillas variádicas en C++. A partir del siguiente programa como ejemplo, deberás crear un programa que utilice plantillas variádicas para realizar una tarea específica.

1. #include <iostream>

2.

3. template <typename T>

4. T min(T a, T b)

5. {

6. return a < b ? a : b;

7. }

8.

9. template <typename T, typename... Args>

10. T min(T a, Args... args)

11. {

12. return min(a, min(args...));

13. }

14.

15. int main()

16. {

17. std::cout << "min(42.0, 7.5)=" << min(42.0, 7.5) << '\n';

18. std::cout << "min(1,5,3,-4,9)=" << min(1, 5, 3, -4, 9) << '\n';

19. }

20.

El programa utiliza una plantilla variádica para calcular el mínimo de una lista de valores. Tu tarea consistirá en diseñar y programar una plantilla variádica que realice una tarea diferente, siguiendo los siguientes requisitos:

1. Tu plantilla debe tener al menos dos parámetros de plantilla: uno para el tipo de datos y otro para el tipo de operación.
2. Tu plantilla debe aceptar un número variable de argumentos del tipo especificado en el primer parámetro de plantilla.
3. Tu plantilla debe realizar la operación especificada en el segundo parámetro de plantilla sobre todos los argumentos y devolver el resultado.

Ejemplos de posibles tareas para tu plantilla podrían ser:

* Calcular la suma, la media o la desviación estándar de una lista de valores.
* Realizar una operación matemática específica (como la multiplicación, la exponenciación, la raíz cuadrada, etc.) sobre una lista de valores.
* Encontrar el valor máximo, mínimo, la mediana o la moda de una lista de valores.
* Concatenar una lista de cadenas de texto.

Ejemplos de posibles entradas y salidas para tu programa podrían ser:

1. // Ejemplo 1: Suma de una lista de valores

2. std::cout << "sum(1, 2, 3, 4, 5) = " << sum<int, std::plus<int>>(1, 2, 3, 4, 5) << '\n'; // Debe imprimir "sum(1, 2, 3, 4, 5) = 15"

3.

4. // Ejemplo 2: Concatenación de una lista de cadenas de texto

5. std::cout << "concat(\"Hola\", \" \", \"mundo\", \"!\") = " << concat<std::string, std::plus<std::string>>("Hola", " ", "mundo", "!") << '\n'; // Debe imprimir "concat(\"Hola\", \" \", \"mundo\", \"!\") = Hola mundo!"

6.