PISTAS.

Pista 1: Definir la función producto

* Definir una plantilla de función variádica que tome un primer argumento de tipo T y cualquier cantidad de argumentos adicionales de tipo Args
* Inicializar una variable de tipo T con el valor del primer argumento
* Utilizar una expansión de paquetes de parámetros para multiplicar la variable por cada argumento adicional
* Devolver el resultado

1. template <typename T, typename... Args> T producto(T a, Args... args) { T result = a; (result \*= args, ...); return result; }

Pista 2: Probar la función producto

* Llamar a la función producto con diferentes tipos de argumentos y comprobar que devuelve el producto correcto
* Asegurarse de probar la función con al menos un argumento

1. int i = producto(1, 2, 3, 4, 5); // i = 120 double d = producto(1.5, 2.5, 3.5, 4.5); // d = 66.375

Pista 3: Definir la función sum\_of\_products

* Definir una plantilla de función variádica que tome un primer argumento de tipo T y cualquier cantidad de argumentos adicionales de tipo Args
* Inicializar una variable de tipo T con el valor del primer argumento
* Utilizar una expansión de paquetes de parámetros para calcular el producto de cada par de argumentos consecutivos y sumar los resultados
* Devolver el resultado

1. template <typename T, typename... Args> T sum\_of\_products(T a, Args... args) { T result = producto(a, args...); if constexpr (sizeof...(args) > 1) { result += sum\_of\_products(args...); } return result; }

Pista 4: Probar la función sum\_of\_products

* Llamar a la función sum\_of\_products con diferentes tipos de argumentos y comprobar que devuelve la suma de los productos correcta
* Asegurarse de probar la función con al menos dos argumentos

1. int i = sum\_of\_products(1, 2, 3, 4, 5); // i = 55 double d = sum\_of\_products(1.5, 2.5, 3.5, 4.5); // d = 38.75

Pista 5: Combinar ambas funciones en la función main

* Llamar a la función sum\_of\_products con diferentes tipos de argumentos y mostrar el resultado
* Asegurarse de probar la función con al menos dos argumentos

1. int main()

2. {

3. std::cout << "Producto de 1, 2, 3, 4, 5: " << producto(1, 2, 3, 4, 5) << '\n';

4. std::cout << "Producto de 1.5, 2.5, 3.5, 4.5: " << producto(1.5, 2.5, 3.5, 4.5) << '\n';

5. std::cout << "Suma de los productos de 1, 2, 3, 4, 5: " << sum\_of\_products(1, 2, 3, 4, 5) << '\n';

6. std::cout << "Suma de los productos de 1.5, 2.5, 3.5, 4.5: " << sum\_of\_products(1.5, 2.5, 3.5, 4.5) << '\n';

7. return 0;

8. }

9.

10.