

PROGRAMACION FUNCIONAL
RODRIGUEZ MORENO MARCO ANTONIO
Aarón Hernández García

Actividad 3.2

01/02/2025

Principales diferencias entre C++ y Haskell

Paradigma de Programación

- C++: Lenguaje imperativo y orientado a objetos. Se ejecuta paso a paso, modificando variables en memoria.
- Haskell: Lenguaje funcional puro. Usa expresiones matemáticas y evita modificar el estado del programa (inmutabilidad).

Estructura y Control de Flujo

- C++: Utiliza un switch para manejar las operaciones y solicita los datos de forma secuencial.
- Haskell: Usa recursión con la función loop, en lugar de estructuras de control tradicionales como while o for.

Entrada y Salida de Datos

- C++: Usa cin y cout de manera directa.
- Haskell: Usa getChar para capturar una tecla y getLine para leer números como texto, convirtiéndolos con read.

Manejo de Errores

- C++: Verifica la división por cero dentro del switch, pero no maneja excepciones explícitamente.
- Haskell: Usa error en caso de divisiones por cero o una opción inválida, lo que puede interrumpir el programa.

Operaciones Soportadas

- C++: Soporta suma, resta, multiplicación y división.
- Haskell: Agrega potenciación ($x ** y$) y raíz cuadrada ($\text{sqrt } x$), permitiendo más funciones matemáticas.

Gestión del Bucle

- C++: Se ejecuta una sola vez y termina. No tiene un bucle interno que permita continuar sin reejecutar el programa.
- Haskell: Usa recursión (loop) para permitir varias operaciones sin necesidad de reiniciar manualmente.

Interfaz y Usabilidad

- C++: Pide los números y la operación en un solo flujo sin opciones visuales de menú.
- Haskell: Usa un menú interactivo, donde el usuario elige una opción y puede salir con la tecla ESC.