

¿Cree que su uso es adecuado en este programa?

Sí, considero que el uso de *JUnit* es altamente adecuado para este programa, ya que permite validar de manera sistemática y automática el correcto funcionamiento de sus componentes principales. En primer lugar, el programa implementa manualmente diversas estructuras de datos fundamentales, como *SinglyLinkedList*, *DoublyLinkedList* y sus respectivas implementaciones en pilas (*ArrayListStack*, *VectorStack*, *ListStack*), por lo que resulta esencial comprobar que la manipulación de nodos y punteros funcione correctamente, especialmente en casos límite como eliminar el último nodo o modificar la cabeza y la cola, evitando así errores como *NullPointerException*. En segundo lugar, el sistema incluye la evaluación de expresiones matemáticas y la conversión de notación *Infix* a *Postfix*, procesos que implican múltiples casos especiales como la precedencia de operadores, el manejo de paréntesis anidados o la división por cero; mediante pruebas unitarias o un enfoque de *Test-Driven Development*, *JUnit* asegura que cada nueva funcionalidad no afecte la lógica previamente validada. Finalmente, al tratarse de un diseño modular orientado a objetos que utiliza interfaces, clases abstractas y patrones como *StackFactory* y *Singleton*, contar con una batería de pruebas facilita la mantenibilidad y la refactorización del código, promoviendo buenas prácticas y mayor confiabilidad en el desarrollo.