Haskell Programming Assignment

1. Write a Haskell function factorial that calculates the factorial of a given integer n. Ensure the function handles edge cases like 0 and negative numbers.

2. Implement a Haskell function isPrime that determines whether a given positive integer n is a prime number or not.

3. Create a Haskell function fibonacci that generates the nth Fibonacci number using a recursive approach. Make sure to handle edge cases, such as when n is 0 or 1.

4. Write a Haskell function reverseList that reverses a list (or a string) using recursion. For example, reverseList [1, 2, 3] should return [3, 2, 1].

5. Implement a Haskell function isPalindrome that checks if a given string is a palindrome (reads the same forwards and backward), ignoring spaces and case sensitivity. For example, "A man a plan a canal Panama" should be considered a palindrome.

Submission Instructions: Haskell Programming Assignments

- File Name: Save your Haskell code in a file named [YourName]_Assignment[X].hs (e.g., AditiSingh_PAssignment.hs).

- Documentation: Include comments explaining your code. Make it easy for the grader to understand your approach.

- Organization: Keep related functions in the same file and label them clearly. Use proper indentation.

- Testing: Include test cases within your code to demonstrate that your functions work correctly.

- Readability: Write clean, well-structured code with meaningful variable and function names.

- Format: Submit your code as plain text files (with a .hs extension), not as binary or compiled files.