INFO1905- Assignment 1- Written Report

Submit a report containing the following:

* Testing report (1 page, plus appendix): Explain succinctly how you tested your code and why your testing is comprehensive. Provide a list of your test cases in the appendix, with expected and observed output.
* Group work (If you submitted as a pair) Explain how you collaborated and the percentage of  authorship for the various sections.

# Testing Report:

The testing process for the project was primarily performed after the various functions were implemented, although the basic tests supplied gave the project some of the benefits that a testing-oriented focus can provide. Basic implementations could be tested for correctness throughout the programming stage, but more specific and complex tests were written after the bulk of programming had concluded.

The project benefited from a comprehensive testing process which contained many of the different test types:

* **Regression tests**: These are added to ensure that any test cases used to define the function (for example, when an empty array is added, or the tree is of an incorrect type) are still met after the code is implemented and updated
* **Tests for invalid input:** These tests are designed to ensure the program handles unexpected input gracefully (within the assumptions provided by assignment description).
* **Tests for edge cases**: Testing the boundary cases to ensure they are handled appropriately ensured the code was handling its logic correctly, and avoided the errors that are often harder to spot.

# Group Work:

Group work for the assessment was undertaken with the goal of doubling the project’s efficiency, as well as providing the opportunity for additional perspectives in regards to test cases and algorithm design.

The main workload was distributed in a 50/50 ratio as follows:

* Katharine:
  + Part 1 (100%)
  + Part 2 (~50%):
    - height( ), height(maxDepth), numLeaves( ), isBinary( ), isProperBinary( )
  + Part 3 (~50%):
    - add( )
* Andrew:
  + Part 2 (~50%):
    - numLeaves(int depth), numPositions(int depth), isCompleteBinary( ), isHeap( ), isBinarySearchTree( )
  + Part 3 (~50%):
    - remove( )
  + Part 4 (100%)

Code collaboration was achieved via a private GitHub repository, with code changes pulled from and pushed to the server as changes were made. Part 5 was completed in collaboration.