# Lab 2 Test Plan

## Black-Box testing techniques used

As black-box testing is being used, we will choose to adopt the most widely used technique for black-box testing which is known as “Boundary Value Analysis”. This will test how the system performs when providing extreme values in the test data and will help to identify any issues or flaws the system may have when such extreme data inputs are provided.

Secondly, we will plan to use “Equivalence Class Partitioning” which is another black-box testing technique. This technique involves checking the input and output by dividing the input into equivalent classes, e.g. a class containing solely valid input values for testing, a class containing invalid test inputs for lower values than expected and another class containing invalid test inputs for higher values than expected.

## How test cases were derived

In order to derive the test cases, prior reading and understanding of the supporting documentation will be required to ensure that the tester understands what the method is intended to do so expected outcomes can be determined ahead of performing the tests. For each Class Under Test, 5 methods from each class will undergo testing and for each of these methods, test cases will be derived by looking at the supporting documentation and designing appropriate test cases based on understanding the method and what it is designed to do alongside understanding the expected and unexpected input values and expected outcomes of each.

## Organization of JUnit test suites

As there are 2 different classes to be tested within the JFreeChart library, the JUnit test suite will be split into 2 sections whereby each class under test will have it’s own section.