**Requirements on the development and maintenance process**

*Quality control procedures*

A J-Unit test suite must be completed upon each module before said module can be considered complete. Testing of a module is only complete if all methods are tested, and each method passes 100% of test cases. Test cases of a method must cover all possible actions which could result from running said method. All test suites must execute within reasonable time (< 1 second) in order for the test to be considered a pass.

* To improve efficiency and prevent overwriting code, any hidden module will assume it will receive input which works as expected, and as such, inputs which would not be received by the module will not be tested. Every assumption made in these hidden modules must be noted explicitly in the testing process however.
* Any module which is available to the external user must handle all possible inputs from the user, including non-sensical inputs. This result can be obtained either through exception handling, or ideally, by preventing the user from inputting non-sensical values at all.

*Priorities of the required functions*

Basic functionality is valued above completeness of the user interface. Therefore, direct user interface modules are to be considered of lowest priority. Next lowest in priority is dataset input methods, due to them being simple, and easy to implement by other coders, while still being important to the workings of the application. Finally, of highest priority is the searching and graph traversal methods. These functions are difficult and highly specific to this application, therefore they must be completed in order for the the project to be considered complete.

*Likely changes to system maintenance procedures*

Execution time required for a code to be considered reasonably fast is an arbitrary value. In certain situations it could grow or shrink with greater evaluation. In addition, the order of the test elements is expected to stay largely static, however, it is possible if due to unforeseen circumstances, a certain method, or module is required in the short term, significantly more than those which have agreed upon higher priority, then the order of priority can shift to represent this.

*Other requirements*

Code should be designed for maintenance with little to no repetition of similar or identical code.