**Guidelines for the implementation of a plan into code, testing and debugging, and documentation of the process**

**CODE**

**SYNTAX**

**RUN-TIME**

**SEMANTIC**

**DOCUMENT**

1. In a new word document, copy the question that the plan attempts to solve. (Format appropriately with heading, header of your name, footer with page number etc.)
2. Develop a table of expected test data.
3. Code the first part of the plan. Ensure that the first two lines include print statements of name of program, author, date and version of code. Include code comments where necessary. Save your code!!
4. Run it. When your program requires test data - **use data from your expected input test data** table.
5. If there are errors (syntax, runtime or semantic) - make a screen dump of them and copy into your documentation document (docD).
6. Correct your code. Run it.
7. When correct, make a screen dump of the output, copy into docD, and copy the correct code into docD, else repeat step 5 and 6.
8. Code the next part of the plan for the program. **Change the version number** in the first few print statements.
9. Repeat steps 5,6 and 7.
10. When your program requires test data - **use data from your expected input test data** table. Make screen dumps of the results from your program.
11. Write comments on whether results are correct or not. If incorrect, explain what you did to correct your code.
12. Copy the improved code to your docD
13. Develop a table of **test data for boundary cases**.
14. Test your code against this data. Modify your code where necessary. Document!!!
15. Develop a table of **invalid test data**. (Out of range?? Invalid data type?? Etc)
16. Test your code against this data. Modify your code where necessary. (loops for data validation?? If statements for insufficient data?? Anything else??) Document!!!
17. Check that your document is well formatted, has page numbers, your name as a header, and that the assessor has access to your program code.