Your submission must be in the form of **a single typeset pdf document**. Note that all standard packages (Word/OpenOffice etc) will produce pdf.

Non-pdf documents will score zero marks.

The document should include **a title, the module code, your name and username**.

The document is to be submitted online at the link below.

The report should describe:

**i. The modular structure of your code (5 marks)**

A brief outline of your code structure:

 - A list of modules for your application with one sentence describing each.

-  For each module: a list of functions in that module with one sentence describing each.

-  A list of header files with one sentence describing each or referring to the appropriate source-code module.

**ii. Your project (10 marks)**

A clear and concise description of your chosen project and what you have implemented.

A clear description of your design strategy:

- how did you plan your work at a high level?

- what were the design iterations you used?

- what were the specific application features associated with each iteration?

**iii. Test cases and results  (10 marks)**

You should describe **five** non-trivial test cases that you have used as part of your application testing.

These should be distinct tests for different functionalities of your code (eg. testing a sorting algorithm is one test, testing three sorting algorithms is still one test)

In each case the test data and inputs that are required, the expected output and actual results.

This should not be the only testing you do as part of your work. It will be assumed that basic testing has been performed during development.

**iv. Personal reflection (5 marks)**

Write one paragraph on each of the following (maximum one a4 side in total).

(i) What went well with this project? Include specific areas of the work, programming, design or testing.

(ii) What was the hardest part of this work? Why, and what will you do to address this for the future?

(iii) What have you learned from this process that you can apply to future programming work in C and other languages?

**Hint**

Please avoid generic statements about time-management. Focus on your C coding, design and testing processes.