# **Assignment 1**

This assignment is to compile a portfolio from the ABM practicals. Submit either a zip file containing all the relevant components to Minerva or a link to your GitHub repository.

For the highest marks, it is expected that there will be a folder for each of the 9 practicals and that each of these practicals will have been completed including the additional tasks outlined in ABM5, ABM6 and ABM7.

Source code documentation/comments are expected to be detailed and extensive and ideally should make it clear what the code does. Each class, function and variable should be described. It is expected that there will be docstrings for functions - listing positional arguments and detailing any returned values.

For a pass mark (50+), you must have completed everything up to ABM5 Section 5. For a merit mark (60+) you must have completed everything up to ABM6 Section 4. For a distinction mark (70+) you must have completed ABM7. For a super distinction mark (80+) you must have completed ABM9.

Your submission will also be assessed against the following criteria which will be used by the assessor to make a judgement on the overall quality of the work and provide a grade and provide you with some brief written feedback that explains the mark and what if anything you might have done better:

**Readability**: The ease with which your code can be understood. Usually helped by: an abundance of clear, concise, informative source code comments; the use of naming conventions; a consistent and standard source code layout (achieved via standard and consistent code indentation, blank lines and the use of standardised documentation syntax); and producing appropriate ancillary documentation as part of the submission.

**Structure**: The degree to which the code has been organised into relevant blocks, files and other structures as appropriate.

**Validity**: The severity and quantity of any logical or functional errors and the presence of appropriate tests and internal checks. (Evidence of testing can be provided in documentation as well as in source code.)

**Efficiency**: The extent to which the code reduces unnecessary computation and memory usage.

**Functionality**: The overall functionality and usability of the software.