L2 Computational Physics Style Guide

Purpose of this guide - this document presents a series of rules that you are expected to follow in your Computational Physics Weekly Assessments. These rules exist to both help you to write and debug your code and help the demonstrators and markers to provide you with useful feedback.

Code shown within an orange dashed rectangle is a transcript of an interactive interpreter session – so this includes code that you type in and the responses as displayed on the interpreter. *You should try entering all examples and running them, both with this style guide and the weekly assessments.*

```
>>> print "Hello world"
Hello world
```

Rule 1 – Python version. The course and associated materials are tested on Python version 2.6 and 2.7. The course requires recent versions of the 'numpy' and 'matplotlib' packages. Python 2.7.5 and the required packages are available through the CIS Windows system, which may be accessed from classroom computers. Alternatively you may use your own machine, in which case the free version of 'Enthought Canopy' is recommended. **However you should ensure your code still works on the CIS Windows system to ensure it can be marked.** Do not use Python version 3.0 or later.

The IDLE environment on the CIS Windows system can be used to both edit, save and run code. To start IDLE, go to the "Start menu" and navigate to "All Programs > Academic Software > Physics > Python 2.7.5 > IDLE (Python GUI)." Python 2.7.5 is also available at the command prompt with the command "python".

Rule 2 – Enabling floating point division. Always use the statement 'from __future__ import division' to instruct Python to use floating point division.

```
>>> print 1/2 # Default integer division
0
>>> from __future__ import division # enable new behaviour
>>> print 1/2 # New floating division
0.5
```

Rule 3 – module import. Unless instructed otherwise, you should only ever import two modules in any weekly assessment submission; namely numpy and matplotlib.pyplot. In external code you may see pyplot and pylab used interchangeably within the matplotlib package. pyplot is the preferred choice. **Do not use non-standard names for imported modules.** The imports are to be used as follows:

```
>>> import numpy
>>> import matplotlib.pyplot as pyplot
```

Rule 4 – variable names. A sensible choice of variable names can impart great clarity and legibility to code, for example using 'ix' for 'index over x' and 'iy' for 'index over y'. Conversely, 'bob', 'bobber', 'bobbob', 'bobby', 'bobbery' and 'bobbert' are often poor choices, especially when used in conjunction. **Good, descriptive variable names help the demonstrators to help you.**