**Assignment 3 Report-Aaron Berryman**

Link to repository: <https://github.com/aberrypicker/PH510>.

Main files involved are ‘monte\_carlo\_class.py’, ‘gaussian\_class.py’ and ‘shape\_points\_class.py’. These contain the code which makes up the classes used in the main code file and contain documentation guiding the reader through their processes. The main code file is ‘assignment\_3\_code.py’, which imports the classes and uses them to initialise the required cases for the task criteria, as well as record the time taken to run these processes for parallel efficiency purposes. The ‘assignment\_3\_slurm.sh’ file contains the code which runs the main file for each different number of cores. This leads to the output file ‘assignment\_3\_code.out’ which contains the determined values, and all the runtimes which can be used to determine the parallel efficiency.

Worked with: Eamonn M., Ben W. (Help with random number generation, bug fixes and Monte Carlo integration/parallelisation.)

The Pluma python 3 file runs all tasks successfully in version 3.12.8 and mpi version gcc-8.5.0/4.1.1, with a pylint score of 10.0/10.0.

On last calculation, using a runtime of 317 seconds for 1 processor, 103 seconds for 2 processors, 57 seconds for 4 processors, 25 seconds for 8 processors and 20 seconds for 16, the parallel efficiency is calculated at 99% (supposedly).

An attempt was made to perform importance sampling on the datasets involved with the Monte Carlo, by attempting to focus on the values of which the function is non-zero or at least near to this value. This was attempted using np.linalg.norm to attempt to create functions to represent q(x), used in importance sampling to represent the term which multiplies the original function. Although I have not achieved it successfully, if I had it should have affected the variance and therefore the uncertainty by reducing them, by being a more accurate representation of the function versus the regular Monte Carlo. This would be most effective for larger calculations and distributions such as the 6D Gaussian case where the variance increases rapidly as the distribution width increases.