**Assignment 2 Report-Aaron Berryman**

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

Worked with: Eamonn M., Ben W., Natalia K. (Reminders on spherical coordinate conversion & behaviours, docstring behaviours, class vs subclass calling, dot & cross product interactions.)

The spyder python file runs all tasks successfully in version 3.11.9, with a pylint score of 10.0/10.0.

The code creates an object class for a vector, defining the initial conditions for it, and then defining the vector interactions including add, subtract, magnitude, dot & cross product, and how it is represented when printed. These are demonstrated by printing the results, where the addition, subtraction, and cross product create new instances of the vector object. The dot product does not create a new instance as it returns a single value. Then a subclass is created to define a Spherical Polar vector and represent it with r, theta, and phi. A conversion routine is placed in the mother class to accommodate the routines when adding/subtracting/dot/cross product between two Spherical Polar vectors in the subclass, as it was found to be the easiest way to perform the calculations. The spherical polar vectors are converted to cartesian, the calculation is performed, and then the resulting vector is reconverted into Spherical Polar coordinates. Guiding docstrings are present for each class and definition within.

A function is created and defined to perform the calculation of the area of the triangle formed between 3 vectors, which determines the vectors of the triangle’s vertices and utilises the half parallelogram area via cross product method to work out the triangle’s area and is performed for all 4 cartesian coordinate triangle systems and all 4 Spherical Polar coordinate systems.

A similar function is defined for the internal angles of the above triangles, using the dot product and magnitude of the vertice vectors to calculate them. Again performed for all 8 triangles.