MTRE 2610 – Intermediate Programming for Mechatronics

Homework – Python Basics

1. Create a 4×4 matrix containing random integer values between 3 and 12 using `numpy.randon.randint`. Display matrix and the sum of all its elements. Replace the values of the 2×2 matrix in the center of the 4×4 with zeros. Display the modified 4×4 matrix.

2. Approximate the value of π by considering a quarter unit circle enclose inside a square. The shaded quarter circle area is $\pi r^2/4 = \pi \cdot 1^2/4 = \pi/4$. The area of the circle can be approximated by choosing a large number of random points inside the square and counting the percentage that fall inside the quarter circle. The equation of a circle with unit radius centered at the origin is

$$\left(x - x_c\right)^2 + \left(y - y_c\right)^2 = r^2 = x^2 + y^2 = 1^2 = 1$$

This is rearranged to give

$$y = \sqrt{1 - x^2}$$

so any (x,y) point inside the square is also is inside the black area if

$$y < \sqrt{1 - x^2}$$

The area inside the square is one square unit and so the ratio of points inside the black area represents the area of the black area. Inside a loop, choose a random position inside the square using `numpy.random.random`. Increment a counter if the point falls inside the quarter circle. Dividing the counter by the total points used gives the ratio of the black area to 1 square unit of area. Knowing computed black area to equal $\pi/4$ allows solving for π. Execute the loop a sufficient number of times to obtain an accurate approximate of pi to 3 significant figures.

3. Using a while loop to increment i, determine and display the number of positive integers for which $i^{2.5}$ is less than 1,000,000.