Experiment 2

Deadline: 2024/4/21 23:59

- Programming Projects
  Textbook (version 9) P250 5.39

Task: Estimating π using Monte Carlo.

An interesting way of calculating π is to use a technique known as Monte Carlo, which involves randomization. This technique works as follows: Suppose you have a circle inscribed within a square, as shown in Figure 1. (Assume that the radius of this circle is 1.) First, generate a series of random points as simple (x, y) coordinates. These points must fall within the Cartesian coordinates that bound the square. Of the total number of random points that are generated, some will occur within the circle. Next, estimate π by performing the following calculation:

$$\pi = 4 \times (number\ of\ points\ in\ circle) \ / \ (total\ number\ of\ points)$$

Write a multithreaded version of this algorithm. Create several threads, each of which generates random points and determines if the points fall within the circle. Each thread will have to update the global count of all points that fall within the circle. Once all the threads have exited, the parent thread will calculate and output the estimated value of π. Protect against race conditions on updates to the shared global variable using mutex locks or semaphores. It is worth experimenting with the number of random points generated. As a general rule, the greater the number of points, the closer the approximation to π.

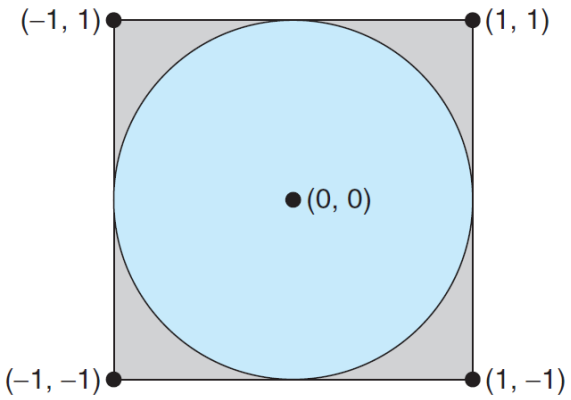

Figure 1

Notes:

(1) Please submit all your source codes.

(2) Please submit your report, showing your experimental procedures and results. Anything else that you would like to report correlated to this project is also welcomed.

(3) You may refer to any other materials besides the textbook for completing the tasks, and please cite them properly in your report.