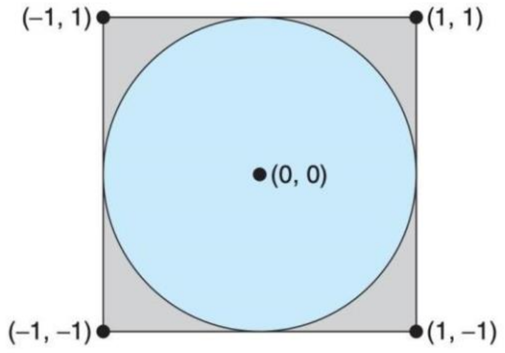
Wayne State University

CSC 4421 – Winter 2025
Computer Operating Systems Labs
Lab 08 – Mutexes in process synchronization

Points possible: 70

Task:

An interesting way of calculating pi 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 below (Assume that the radius of this circle is 1):



The given sample program (sample.c) generates 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 by performing the following calculation:

pi = *4 × (number of points in circle) / (total number of points)*

The given program creates *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. You must modify the program to protect against race conditions on updates to the shared global variable by using *mutex locks*.

Sample Execution:

```
$. ./out 5
Pi is: 2.400000

$. ./out 50000  Pi
is: 3.141440
```