SISTEMAS OPERATIVOS

Tecnológico de Monterrey, Campus Querétaro

Laboratorio 7 - Hilos

1. Ejercicio 1

Una forma interesante de calcular π es usar una técnica conocida como Monte Carlo, que implica la aleatorización. Esta técnica funciona de la siguiente manera: supongamos que tienes un círculo inscrito dentro de un cuadrado, como se muestra en la Figura 1. Supongamos que el radio de este círculo es 1. Primero, generamos una serie de puntos aleatorios (x, y). Estos puntos deben estar dentro de las coordenadas cartesianas que unen el cuadrado. Del número total de puntos aleatorios que se generaron, algunos ocurrirán dentro del círculo. Luego, calculamos π realizando el siguiente cálculo:

$$\pi = \frac{4 * number\_in\_circle}{number\_of\_tosses} \tag{1}$$

1. Escribe una versión multihilo en Java de este algoritmo que crea un hilo separado para generar una cantidad de puntos aleatorios. El hilo contará la cantidad de puntos que ocurren dentro del círculo y almacenará el resultado en una variable global. Cuando este hilo ha salido, el hilo padre calculará y generará el valor estimado de π. Experimenta con la cantidad de puntos aleatorios generados. Como regla general, cuanto mayor sea el número de puntos, más cercana será la aproximación a π.

```
number_in_circle = 0;
for (toss = 0; toss < number_of_tosses; toss++) {
  x = random double between -1 and 1;
  y = random double between -1 and 1;
  distance_squared = x*x + y*y;
  if (distance_squared <= 1) number_in_circle++;
}
pi_estimate = 4 * number_in_circle/((double) number_of_tosses);
```

2. Ahora, desarrolla la versión multihilo pthread.

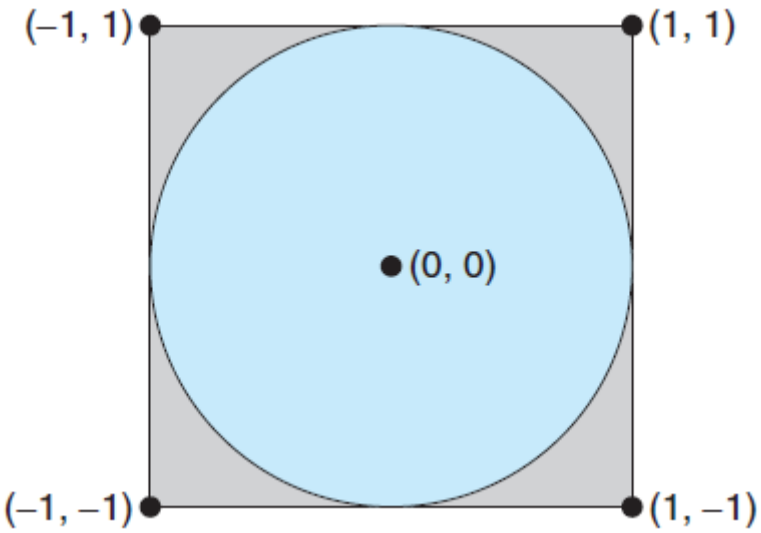

Figura 1: Técnica de Monte Carlo para calcular π.