

# Thread Safety

Computational Science II (CAAM 520)

---

Christopher Thiele

Rice University, Spring 2020

## Motivating example

Let us compute an approximation to  $\pi$  using a Monte Carlo method:

Generate random points in  $[0, 1]^2$  and count the points inside the quarter circle given by

$$\sqrt{x^2 + y^2} \leq 1.$$

Then

$$\frac{\pi}{4} \approx \frac{\#(\text{points inside quarter circle})}{\#(\text{points})}.$$

## Motivating example

How can we parallelize the computation using OpenMP?

## Motivating example

How can we parallelize the computation using OpenMP?

Is our initial attempt correct?

## Motivating example

How can we parallelize the computation using OpenMP?

Is our initial attempt correct?

→ No, because `rand()` is not ***thread safe***!

# Thread safety

A function is called ***thread safe*** if it is guaranteed that no race conditions occur when calling the function concurrently from multiple threads.

How do we know whether a function is thread safe?

→ Consult its documentation.

# Thread safety

A function is called ***thread safe*** if it is guaranteed that no race conditions occur when calling the function concurrently from multiple threads.

How do we know whether a function is thread safe?

→ Consult its documentation.

Back to our example: Why is `rand()` not thread safe?

# Thread safety

**Note:** In general, a thread safe function may still result in

- poor performance, or
- deadlocks

when called concurrently.