D-BAUG Informatik I

Exercise session: week 3

HS 2018

Homework

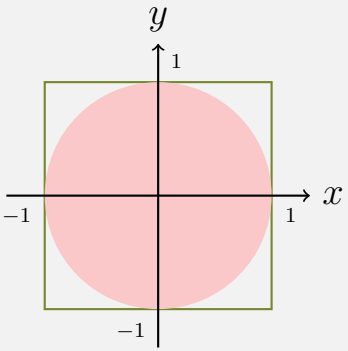
- Questions?

Monte Carlo Simulation

Monte Carlo Simulation: Use randomness to solve problems. Wide area of applications in applied mathematics, all natural sciences and engineering

Estimate π using Monte Carlo Simulation

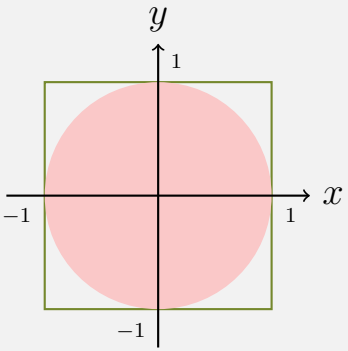
$$\frac{\text{circle area}}{\text{square area}} = \frac{\pi}{4}$$



Estimate π using Monte Carlo Simulation

$$\frac{\text{circle area}}{\text{square area}} = \frac{\pi}{4}$$

Idea: simulate random variable with uniform distribution on the unit square $[0, 1] \times [0, 1]$.

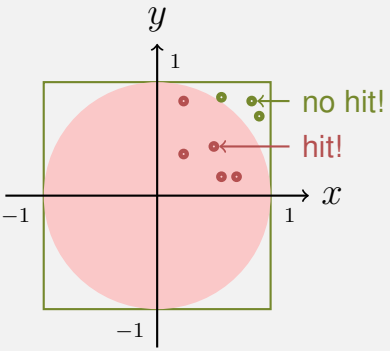


Estimate π using Monte Carlo Simulation

$$\frac{\text{circle area}}{\text{square area}} = \frac{\pi}{4}$$

Idea: simulate random variable with uniform distribution on the unit square $[0,1]\times[0,1]$.

$$\frac{\text{number hits}}{\text{number trials}} \cdot 4 \approx \pi.$$



Task

- Conduct an experiment: estimate π using the Monte Carlo method above.
- Execute the experiment for number of trials 1, 2, 4, 8, $\ldots$, n, n should be asked from the user.
- Output the number of trials and the estimation of π each time.

Hint: a pseudo-random number uniformly disrtibuted on $[0, 1)$ can be obtained via `Math.random()`.

Extension

Deterministic formulas to approximate π:

$$Sum_1 : \frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots$$
$$Sum_2 : \frac{\pi}{2} = 1 + \frac{1}{3} + \frac{1 \cdot 2}{3 \cdot 5} + \frac{1 \cdot 2 \cdot 3}{3 \cdot 5 \cdot 7} + \cdots$$

Compare the approximation accuracy of the different methods.

Solution on next slide

spoiler ahead – do not open (yet)

Solution on next slide

spoiler ahead – do not open (yet)

Estimate π using Monte Carlo Simulation

```
public class Main {
  public static void main(String[] args){
    int n = In.readInt();
    for (int trials = 1; trials <= n; trials*=2){
      int hits = 0;
      for (int i = 0; i<trials; ++i){
        double x = Math.random();
        double y = Math.random();
        if (x * x + y * y <= 1){
              hits++; }
      }
      double pi = (double)hits / trials * 4;
      Out.println("trials=" + trials + ", pi=" + pi);
    }
  }
}
```