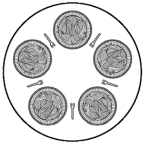
Concurrency

EECS 4315

www.eecs.yorku.ca/course/4315/

The Dining Philosophers Problem

In the dining philosophers problem, due to Dijkstra, five philosophers are seated around a round table. Each philosopher has a plate of spaghetti. The spaghetti is so slippery that a philosopher needs two forks to eat it. The layout of the table is as follows.



The life of a philosopher consists of alternative periods of eating and thinking. When philosophers get hungry, they try to pick up their left and right fork, one at a time, in either order. If successful in picking up both forks, the philosopher eats for a while, then puts down the forks and continues to think.

The Dining Philosophers Problem

```
public class Philosopher {
  private int id;
  private Table table;

  public Philosopher(int id, Table table) {
    this.id = id;
    this.table = table;
  }
  public void run() {
    while (true) {
      this.table.pickUp(id);
      this.table.pickUp(id + 1 % 5);
      // eat
      this.table.putDown(id);
      this.table.putDown(id + 1 % 5);
    }
  }
```

The Dining Philosophers Problem

```
public class Table {
  public void pickUp(int id) { ... }
  public void putDown(int id) { ... }
}
```

The Dining Philosophers Problem

Question

Solve the problem.

Race condition and data race

A race condition is a flaw that occurs when the timing or ordering of events affects a program's correctness. Generally speaking, some kind of external timing or ordering non-determinism is needed to produce a race condition.

A data race happens when there are two memory accesses in a program where both

- target the same location,
- are performed concurrently by two threads,
- are not reads (at least is a write),
- are not synchronization operations.

Race condition and data race

Many race conditions are due to data races, and many data races lead to race conditions. However, we can have race conditions without data races and data races without race conditions.

Race condition and data race

Many race conditions are due to data races, and many data races lead to race conditions. However, we can have race conditions without data races and data races without race conditions.

Question

Give an example that has both a data race and a race condition.

Race condition and data race

Many race conditions are due to data races, and many data races lead to race conditions. However, we can have race conditions without data races and data races without race conditions.

Question

Give an example that has both a data race and a race condition.

Question

Give an example that has a race condition but does not have a data race.

Race condition and data race

Many race conditions are due to data races, and many data races lead to race conditions. However, we can have race conditions without data races and data races without race conditions.

Question

Give an example that has both a data race and a race condition.

Question

Give an example that has a race condition but does not have a data race.

Question

Give an example that has a data race but does not have a race condition.

Detecting data races with JPF

```
target=Example
classpath=.
listener=gov.nasa.jpf.listener.PreciseRaceDetector
```