Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Parallel Programming
Assignment 7: Supplementary Exercises
Spring Semester 2020

Pipelining

Let us assume that 4 people are at the airport. To prepare for departure, each of them has to first scan their boarding pass (which takes 1 min), and then to do the security check (which takes 10 minutes).

a) Assume that there is only one machine for scanning the boarding pass and only one security line. Explain why this pipeline is unbalanced. Compute its throughput.

b) Now assume that there are 2 security lines. Which is the new throughput?

c) If there were 4 security lines opened, would the pipeline be balanced?

Wait and Notify

Consider the following implementation of a FairThreadCounter which implements the Round Robin policy for 2 threads (as described in exercise 3).

```java
public class FairThreadCounter extends ThreadCounter {

  public FairThreadCounter(Counter counter, int id, int numThreads, int numIterations) {
    super(counter, id, numThreads, numIterations);
    assert numThreads == 2
  }

  public void run() {
    for (int i = 0; i < numIterations; i++) {
      synchronized (counter) {
        counter.increment();
        counter.notify();
        try {
          counter.wait();
        } catch (InterruptedException e) {
          e.printStackTrace();
        }
      }
    }
  }
}

public static void main(String[] args) {
  Counter counter = new SequentialCounter();
  count(counter, 2, ThreadCounterType.FAIR, 10);
  System.out.println("Counter: " + counter.value());
}
```

a) What will be printed in the console after running the program?

b) Does the solution behave as expected? If not, explain why and fix the errors.