2.5 Lecture Summary

2 Functional Parallelism

2.5 Determinism and Data Races

Lecture Summary: In this lecture, we studied the relationship between determinism and data races in parallel programs. A parallel program is said to be *functionally deterministic* if it always computes the same answer when given the same input, and *structurally deterministic* if it always computes the same computation graph, when given the same input. The presence of data races often leads to functional and/or structural nondeterminism because a parallel program with data races may exhibit different behaviors for the same input, depending on the relative scheduling and timing of memory accesses involved in a data race. In general, the absence of data races is not sufficient to guarantee determinism. However, all the parallel constructs introduced in this course ("Parallelism") were carefully selected to ensure the following *Determinism Property*:

If a parallel program is written using the constructs introduced in this course *and is guaranteed to never exhibit a data race*, then it must be both functionally and structurally deterministic.

Note that the determinism property states that all data-race-free parallel programs written using the constructs introduced in this course are guaranteed to be deterministic, but it does not imply that a program with a data race must be functionally/structurally non-deterministic. Furthermore, there may be cases of "benign" nondeterminism for programs with data races in which different executions with the same input may generate different outputs, but all the outputs may be acceptable in the context of the application, e.g., different locations for a search pattern in a target string.

Optional Reading:

1. Wikipedia article on Race condition

✓ Complete

Go to next item