23.15 (Advanced) Fork/Join Framework

8

Java's concurrency APIs include the Fork/Join framework, which helps programmers parallelize algorithms. The framework is beyond the scope of this book. Experts tell us that most Java programmers will nevertheless benefit by the Fork/Join framework's use "behind the scenes" in the Java API and other third-party libraries. For example, the parallel capabilities of Java SE 8 streams are implemented using this framework.

The Fork/Join framework is particularly well suited to divide-and-conquer-style algorithms, such as the recursive merge sort that we implemented in Section 19.8. Recall that the recursive merge-sort algorithm sorts an array by *splitting* it into two equal-sized subar-rays, *sorting* each subarray, then *merging* them into one larger array. Each subarray is sorted by performing the same algorithm on the subarray. For algorithms like merge sort, the Fork/Join framework can be used to create concurrent tasks so that they can be distributed across multiple processors and be truly performed in parallel—the details of assigning the tasks to different processors are handled for you by the framework. Exercises 23.20–23.21 ask you to further investigate the Fork/Join Framework and use it to reimplement

the recursive merge sort and quicksort algorithms. Exercise 23.22 asks why you might not want to take the time and effort to reimplement the binary search algorithm with Fork/Join. To learn more about Fork/Join, see the following Oracle tutorial and other online tutorials:

```
https://docs.oracle.com/javase/tutorial/essential/con
```