



Course Certificate

Parallel Programming in Java



[View Certificate](#)



Completed by **Julie Lollis**

December 14, 2019



Four weeks of study, 4-8 hours/week depending on past experience with sequential programming in Java



Julie Lollis's account is verified. Coursera certifies their successful completion of Parallel Programming in Java.

Share this Certificate



<https://coursera.org/share/49521973d9e13c>

Copy Link

About this Course

Taught by: Vivek Sarkar

Rice University

This course teaches learners (industry professionals and students) the fundamental concepts of parallel programming in the context of Java 8. Parallel programming enables developers to use multicore computers to make their applications run faster by using multiple processors at the same time. By the end of this course, you will learn how to use popular parallel Java frameworks (such as ForkJoin, Stream, and Phaser) to write parallel programs for a wide range of multicore platforms including servers, desktops, or mobile devices, while also learning about their theoretical foundations including computation graphs, ideal parallelism, parallel speedup, Amdahl's Law, data races, and determinism.

Why take this course?

- All computers are multicore computers, so it is important for you to learn how to extend your knowledge of sequential Java programming to multicore parallelism.
- Java 7 and Java 8 have introduced new frameworks for parallelism (ForkJoin, Stream) that have significantly changed the paradigms for parallel programming since the early days of Java.
- Each of the four modules in the course includes an assigned mini-project that will



provide you with the necessary hands-on experience to use the concepts learned in the course on your own, after the course ends.



- During the course, you will have online access to the instructor and the mentors to get individualized answers to your questions posted on forums.

The desired learning outcomes of this course are as follows:

- Theory of parallelism: computation graphs, work, span, ideal parallelism, parallel speedup, Amdahl's Law, data races, and determinism
- Task parallelism using Java's ForkJoin framework
- Functional parallelism using Java's Future and Stream frameworks
- Loop-level parallelism with extensions for barriers and iteration grouping (chunking)
- Dataflow parallelism using the Phaser framework and data-driven tasks

Mastery of these concepts will enable you to immediately apply them in the context of multicore Java programs, and will also provide the foundation for mastering other parallel programming systems that you may encounter in the future (e.g., C++11, OpenMP, .Net Task Parallel Library).

[Learn more >](#)

Coursera

About
Leadership
Careers
Catalog
Certificates
MasterTrack™ Certificates
Degrees
For Enterprise
For Government
For Campus

More

[Terms](#)

Community

Learners
Partners
Developers
Beta Testers
Translators
Blog
Tech Blog

[Privacy](#)

[Help](#)



[Accessibility](#)

[Press](#)

[Contact](#)

[Directory](#)

[Affiliates](#)

Learn Anywhere



Follow Us



© 2020 Coursera Inc. All rights reserved.