

PARALLEL COMPUTING IN JULIA

Instructors:

Alan Edelman, MIT Professor of Applied Mathematics and Chief Scientist at Julia Computing

Dr. Matt Bauman, Senior Research Scientist at Julia Computing and long-time open source contributor to the language.

Avik Sengupta, VP of Engineering at Julia Computing, and a long time contributor to the Julia language.

Mode of instruction: Online (Detail will be shared soon)

Parallel computing is hard. Julia makes parallel computing easier. In this training, we discuss modern trends in high performance computing and how they've converged towards multiple types of parallelism. This interactive course demonstrates how to use Julia for distributed computing with or without message passing interface (MPI), shared memory computing including threads and tasks and computing using graphics processing units (GPUs). Julia makes all these modes of parallelism possible, and the Julia community is continuing to perform active research to make high performance parallel computing easier. Many national labs, major corporations, and universities are already using Julia with parallel computing. We will conclude with upcoming future developments including work on Google's tensor processing units (TPUs). We will provide cloud computation resources appropriate for parallelism for the duration of the course.

Course Objectives:

1. Identify the challenges in converting a program from serial to parallel
2. Discover the many forms of parallelism Julia offers and learn when to use each
3. Learn how to structure programs to take advantage of parallel computation
4. Write programs that use each kind of parallelism

The concepts will be taught in Julia, the fastest and most productive modern high-level language for numerical computing and machine learning.

Duration: 8 hours (4 hours each day)

Pre-requisites:

Participants should have basic understanding of non-parallel programming techniques and of Julia itself. Those who are new to Julia are invited to take Introduction to Julia first.