**Blue Waters Petascale Semester Curriculum v1.0**

**Unit 4: OpenMP**

**Lesson 6: When Should You Use OpenMP?**

**Instructor Guide**

*Developed by Colleen Heinemann for the Shodor Education Foundation, Inc.*



*Except where otherwise noted, this work by The Shodor Education Foundation, Inc. is licensed under CC BY-SA 4.0. To view a copy of this license, visit*[*https://creativecommons.org/licenses/by-sa/4.0*](https://creativecommons.org/licenses/by-sa/4.0)

*Browse and search the full curriculum at*[*http://shodor.org/petascale/materials/semester-curriculum*](http://shodor.org/petascale/materials/semester-curriculum)

*We welcome your improvements! You can submit your proposed changes to this material and the rest of the curriculum in our GitHub repository at*[*https://github.com/shodor-education/petascale-semester-curriculum*](https://github.com/shodor-education/petascale-semester-curriculum)

*We want to hear from you! Please let us know your experiences using this material by sending email to* [*petascale@shodor.org*](mailto:petascale@shodor.org)

The examples provided in this lesson provide some preparation by the instructor prior to doing the examples in class. Example #1 in the presentation slides requires that the instructor bring in 7 pieces of paper, each with a number on it from 1 to 7. So, paper one would have a 1 on it, paper two would have a 2 on it, and so on. In Iteration 1 of the example, these are all going to be placed in the same area of the room. In the rest of the iterations, though, the papers need to be spread out across the room. Students will also need paper and pens to write down their results or, if available, they can each be given a designated section of a whiteboard or chalkboard.

It is important to note that, as this is a very difficult topic to grasp, especially for students that have never programmed before, it might be necessary for the instructor to plan out a few different ways to explain the topic being covered.

There are exercises provided with instructions for the students, but it is important that the instructor test the commands and modify the run commands depending if the commands provided do not work for the given supercomputer/cluster that the instructor has access to for the course. The exercises are meant to be as generic with as few dependencies as possible, but it is possible some modifications may need to be made in order for the students to be able to successfully compile and run the programs.

**Common Pitfalls for Students and Instructors**

It is likely that students, especially those with no programming experience, may have some difficulty understanding the concept of shared memory parallelism, or even parallelism in general. Parallelism is a very different way of thinking than we normally think in our day to day lives, so it may take some students longer than expected to grasp the concept. Similarly, finding new and innovative ways for instructors to teach the same ideas might be difficult as well. This lesson attempts to provide several different examples to show different ways of teaching the same concept.