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

**Unit 9: Optimization**

**Lesson 4: Multiprocessor Caching and False Sharing**

**References / Further Reading**

*Developed by* *David P. Bunde 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)

Other examples of false sharing are discussed online, including the following:

* <https://en.wikipedia.org/wiki/False_sharing>
* <https://stackoverflow.com/questions/22766191/what-is-false-sharing-how-to-reproduce-avoid-it>

Our example is different in that it’s a potentially useful program (checking the load balance of the Mandelbrot computation). These examples are nice for being small and emphasizing the effect (having simple other code makes it conceivable that time on the entire program could detect false sharing rather than having to isolate the main loop and measure only that).