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

**Unit 7: CUDA**

**Lesson 8: CUDA Memory Model**

**Instructor Guide**

*Developed by Maria Pantoja 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)

1. (ppt) Slides Contain:
   1. Learning Objectives
   2. Explanation of the problem Sequential
   3. Explanation of how to parallelize the problem
2. Code is written in C
   1. How to compile and run the code is explained directly on the slides with the command line compiler instruction
   2. The environment used :CPU type, GPU type, OS , Compiler version (gcc, CUDA, MPI, OpenMP)
3. Video Lecture:
   1. Narration of the slides
4. Code:
   1. Implementation of the example explained in the slides

**Common Pitfalls for Students and Instructors**

Students:

1. Ignoring the race condition on the hist[]++
2. Confusing Shared and Global memory and try to access the shared memory array (only numberOfThreadsPerBlock wide-256) with the global threadId