UNIVERSITÉ DU LUXEMBOURG

Rule 110: Faster

Parallel Computing

Goals

- ✯ Learn to parallelize a code using C++ execution policy, OpenMP and `std::thread`.
- ✯ **Relevant videos**:
  - Same as the lab "Rule 110: Three Ways to Parallelize", in particular Arithmetic Intensity.
  - False Sharing
  - Stanford videos (lectures 1 to 5).

Information

1. The code on your Github repository generated by clicking here: `https://classroom.github.com/a/2bzpemon`
2. **Reviewer:** Paul Aromolaran (Github: `PaulAroo`)
3. **Video:** Produce a 3 minutes (maximum) explanatory video of your code (provide the link in the README). Record your screen and your voice. You can upload the video on the University onedrive.
4. **Automated leaderboard**

Rules

1. You can discuss your design and your results on Discord or orally, but please don't share your code.
2. This is a solo project.

Exercise 1 – Towards efficiency

Add a new version flag `--version efficient` providing the fastest parallel implementation you can. For us to assess the correctness of your code, print the sum of the array in the last iteration (and nothing else).

Exercise 2 – Benchmarking

Benchmark your code with the different versions, and various size of arrays and iterations. Plot your results and discuss the plots and results in the README.md.

Exercise 3 – With pattern recognition (+0.5 bonus point on the final grade)

Support pattern counting (only for the version `--version efficient`).