teuscher Lab
teuscher-lab.com

Christof Teuscher
ECE 410/510
Spring 2025

Portland State University

Week 4
Challenges
ECE 410/510
Spring 2025

Instructions:

- The challenges below are for you to delve deeper into the subject matter and to test your own knowledge.
- I'd suggest you try to solve at least one problem per week. More is obviously better.
- Practice "vibe coding" if necessary.
- Post your solution(s) in the #weekly-challenges Slack channel so everybody can appreciate what you did, ask questions, and make comments.
- Document everything for your portfolio and make your code available on Github.

Challenge #13: Benchmarking different SAXPY problem sizes.

Learning goals:

- Set up a CUDA development environment.
- Modify existing CUDA code.
- Profile and optimize your implementation.
- Compare and visualize the performance as you increase the problem size.

Tasks:

- Download the example code from https://developer.nvidia.com/blog/easy-introduction-cuda-c-and-c
- Modify the code so that you can simulate and measure the execution times for the following matrix sizes: $N = 2^{15}, 2^{16}, \ldots 2^{25}$ (or as high as you can).
- Visualize the execution times in a bar plot. You can use a spreadsheet or Matplotlib for that. What do you observe?
- If you want to make this even fancier, measure the total execution time (including memory transfers and allocations and the kernel execution time only (just the GPU computation) separately. Hint: `cudaEvent`
- If you don't have access to a GPU, you could use Google Colab:
    - https://colab.google
    - The free version of Colab gives you access to NVIDIA Tesla K80 GPUs.
    - https://www.nvidia.com/en-gb/data-center/tesla-k80

Challenge #14: Fibonacci sequence in CUDA

- Write a CUDA kernel that computes the Fibonacci sequence for N numbers. E.g., $N = 2^{20}$.
- The Fibonacci sequence is a famous mathematical sequence where each number is the sum of the two preceding ones. It starts with 0 and 1, and continues as follows: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ...
- The mathematical definition is:
    - $F(0) = 0$
    - $F(1) = 1$
    - $F(n) = F(n-1) + F(n-2)$ for $N-2 \geq n > 1$
- Compare a simple sequential implementation with your CUDA implementation. What do you observe?