CS205 – High Performance Computing for Science and Engineering

Matheus S. Farias*

School of Engineering and Applied Sciences, Harvard University

Spring, 2022

Abstract

This document is composed by lecture notes of CS205 – High Performance Computing for Science and Engineering[1], taught by Professor Fabian Wermelinger in Spring 2022. I am responsible to all mistakes here written.

Contents

§1 01/25 – Course Overview

Question 1.1. The author means there is a limitation in the Bottom – or semiconductor technology – governed by the laws of Physics. That is, we cannot improve too much in the low-level perspective of efficiency by utilizing the same procedure engineers have been working on the last decades (CMOS-based technology). Now we should shift the aim for the Top, which the author mentions as the triplet: software, algorithm, and hardware architecture.

Question 1.2. Although the Bottom view of technology arised along these ways, real application did not follow the same slope of performance, creating an efficiency gap. This gap is due to the lack of optimal usage of the Top resources available, such as leveraging parallelism through multi-core

*E-mail address: matheusfarias@g.harvard.edu

[1]Link to the class' website: https://harvard-iacs.github.io/2022-CS205/

processing, temporal and spatial locality principles, etc.

Question 1.3. There is room for optimization in the Top, by changing the programming language, leveraging hardware architecture, etc. We can obtain more efficiency in our code. But it is not always a good idea to keep pushing into this optimality once the larger the version the harder is to write and understand code. Sometimes the application does not requires such high-end performance, the engineer just need to evaluate this tradeoff.

Question 1.4. The usage of multi-core processing.

Question 1.5. The memory. Approximately 20×.

Question 1.6. To run multiple operations from a single instruction.

there is not uniform encoding

References