GP257 Syllabus

[Winter 2021]

Instructor Information

Instructor	Email	Office Location & Hours
Bob Clapp	clapp@stanford.edu	Check my calendar

General Information

Description

A course to improve your coding efficiency. The purpose is of this class to efficiently teach you to produce code more efficiently that is efficient.

Expectations and Goals

You can take this course for 1-4 credits depending on the sub-sections you are interested in. You will be required to complete 100 points of labs for each section you signup for at least 50 points in each subsection you sign up for.

Course Materials

Required Materials

You will need access to a computer.

Optional Materials

You will be given websites to look at to provide more information on various topics covered in the course.

Course Sections

Section 1: Python (January 11-January 25)

Starting from the very basics of python from a more numerical scientist perspective. We will start with basic python and writing functions in a basic object-oriented style with unit testing. We will move on how to use pytorch and numba to significantly increase performance.

Section 2: Compiled languages (January 27-February 10)

We will start with how to write portable code using Cmake. We will then move on to how to write optimized serial code and how to test how efficient your code is vs an optimal code. We will finish with a discussion of how to vectorize a code and parallelize a code on a single machine.

Section 3: Massively parallel (February 12-February 26)

We will discuss parallel architectures from clusters to GPUs, to the cloud. We will talk about how to interact with queuing systems, MPI programming, and taking advantage of a GPU using pyCuda and numba.

Section 4: Advanced Topics (March 1- March 12)

We will talk about advanced portability using singularity and docker. We will talk about how to write Cuda codes, how to bind compiled languages into python using pybind11, and we will talk about the current “future” of parallel computing using Kubernetes.

Course format

Each section will involve a series of in-class lectures, programming examples, and in-class discussions. We will have occasional quizzes.

Grading

80% labs
5% in-class participation
15% quizzes