# AM 250

# Introduction to High Performance Computing


**Nic Brummell**

brummell@ucsc.edu
BE359A

(831) 459 2122
Applied Math (AM)
University of California, Santa Cruz

# **AM 250 2023:** Course details

## WELCOME!

People: Me = Nic Brummell

brummell @ ucsc.edu

general oversight of teaching: organization, lectures, homework, questions, office hours, programming help, etc

TA = Winson Chen

wchen157 @ ucsc.edu

Programming help, homework grading, etc

Times: Mon/Wed/Fri 9:20am – 10:25am

Where: McHenry Library 1350

# AM 250 2023: How this class runs

- LECTURES will be
  - *Synchronous and live!*
  - They are also RECORDED (Canvas > YuJa; media.ucsc.edu)
  - They are NOT TELECAST (i.e. NOT broadcasted live)
  - There are essentially two categories of lectures:
    i. Theoretical: motivation, concepts, ideas, models (machine, programming, performance)
    ii. Practical: Machine use, programming
- ASSIGNMENTS:
  - Most of the *real learning* will be done in *practical, out-of-class assignments*
  - This is about computing, so you will have to compute!
  - Many of the hard parts are in the details of dealing with machines so you need *practice!*
  - The concepts of parallel programming doesn't sink in until you *type in a program and try and run it in parallel!*

# AM 250 2023: Office hours

- To help with the lectures and assignents, there will be a weekly OFFICE HOURS given by both me and the TA. These may consist of …
  - Q&A for clarification of ideas and concepts from the lectures
  - Q&A regarding assignments
  - Actual practical help getting things working
  - GOAL: To solidify concepts and provide technical help

- These office hours will generally be IN-PERSON but some might be ONLINE
  - I currently added one for Wed 2pm at myoffice BE 359A
  - TA Winson : Mon 3-4pm & Thurs 9-10am

# AM 250 2023: Websites

Websites:

EVERYTHING IS RUN THROUGH CANVAS!! canvas.ucsc.edu

Modules guide you through the work for each week

Assignments are all set and handed in from Canvas

YuJa stores all the videos of (old and current) lectures are there (or go to media.ucsc.edu and log in)

Zoom gives access to any Zoom links for online meetings etc

Etc

→ Take a quick tour?

NOTE: Another website with lots of GREAT stuff: keeplearning.ucsc.edu

Grading:

6 homeworks/programming assignments

1 final project (just a somewhat more in-depth programming assignment)

# **AM 250 2023**: First activities!

# **AM 250 2023**: Introductions!

**Think – pair – share!**

1. Name
2. Degree program
3. Reasons for being here in class
4. What are you afraid of/apprehensive about/worried about/anxious about with regards to this class?
5. What do you envisage being able to do by the end of this class that you can't do now?

# **AM 250 2023:** Technical survey!

Do the online survey in the Canvas module for Week1!

Quick survey here by show of hands …

Who has experience already with:

- ✓ Unix/Linux environment? (commands like ls, cd, pwd etc + ssh, …)
- ✓ Shell scripting languages? (sh, csh, bash, …)
- ✓ Programming language? (Fortran, C, …)
- ✓ Using parallel environments on supercomputers? (i.e. running codes)
- ✓ Parallel programming (i.e. actually writing code)? MPI? OpenMP?

# What is the point of this course?

Learn how to harness all this computing power!

In particular, learn how to design and use code for **MASSIVELY PARALLEL MACHINES**

**Elusive mixture of theory and practice …**

Highly interdisciplinary! These days need:

- ✓ Engineering skills: architecture, hardware, network knowledge
- ✓ Mathematical skills: applied math methods, numerical analysis
- ✓ Computer science skills: algorithm design, software engineering, compilers, operating systems, ftp, …
- ✓ Scientific skills: design of problem, setup of models/equations, analysis of results
- ✓ Artistic skills: visualisation, movies, …

# The syllabus

PART A: CONCEPTS

- Intro to Parallel Computing
  - ✓ Parallel machine models
  - ✓ Parallel programming models
- Designing Parallel algorithms
  - ✓ Partitioning, communication, agglomeration, mapping
  - ✓ Performance analysis

PART B: TOOLS

- Environment: Unix; batch systems; svn, github?
- Programming: Fortran?
- Programming: MPI, OpenMP, …
- Debugging, performance analysis
- Analysis ?(IDL?, Python, VAPOR)

PART C: SPECIALTY ITEMS?

- Parallel Math libraries; Parallel IO
- Manycore: GPU, Intel Phi?
- Map-Reduce, Hadoop, …

PART D: CASE STUDIES?

- Spectral Methods (HPS); Finite-Volume; N-Body?

**EASY!**

**But very seldom all collected together in one place for you!**

# Learning outcomes

A pretty in-depth understanding of:

- When and why parallel computing is useful
- Parallel computing hardware options
- Parallel programming models + some associated software and tools
- Some basic experience using examples of hardware and software
- Knowledge of debugging, performance analysis and tuning
- Knowledge of some important parallel applications and their algorithms
- Exposure to some open research questions

# Expectations

*Expectations of the student:*

- ✓ Try to learn something useful!
- ✓ There will be exercises and tasks: try to do them!

*Intention of the course is that it is a practical introduction.*

*You therefore have to try. The struggle is real, and is useful!*

Note also that:

- ✓ *There are very seldom "right" and "wrong" answers*
- ✓ *Often it is possible that I won't know the answer!*

# Books, resources …

Beyond lecture materials: (see resources in Canvas pages)

Part 1

- Much based on Ian Foster's (ANL) excellent analysis book "Designing and Building Parallel Programs" available for free on the web:

  http://www.mcs.anl.gov/~itf/dbpp/text/book.html

- Supplemental textbooks:
  - "Programming on Parallel Machines", by Norm Matloff, UC Davis

    http://heather.cs.ucdavis.edu/parprocbook
  - "Structured Parallel Programming: Patterns for Efficient computation", by Michael McCool, Arch Robinson, James Reinders & Morgan Kaufmann

    http://www.sciencedirect.com/science/book/9780124159938
  - "Introduction to Parallel Computing", by Ananth Grama, Anshul Gupta, George Karypis & Vipin Kumar

    http://www-users.cs.umn.edu/~karypis/parbook/

Part 2: MPI, OpenMP

- LOTS of tutorials on web
- Some useful reference books too, but really probably not worth buying anything

# **AM 250 2023:** Another baseline TEST!

# AM 250 2023: Another baseline TEST!

Expand the following acronyms:

HPC
CPU
PE
GPU
GPGPU
SISD
SIMD
MISD
MIMD
SMP
UMA
NUMA

MPP
MIPS
FLOPS
MFLOPS
GFLOPS
TFLOPS
PFLOPS
LAN
WAN
MPI
PVM

CD
LS
SSH
CVS
SVN
FFT
DNS
LES
NHB

Explain the following concepts:

Vector processing
Parallel processing
Shared memory
Distributed memory
Threads
Concurrency
Scalability
Modularity
Determinism
Load balancing
Data dependency
Embarrassingly parallel
Prefetching
Partitioning
Domain decomposition

Agglomeration
Butterfly
Divide and conquer
Mapping
Latency
Bandwidth
Amdahl's Law
Women are from Venus, Mean are from Mars …