Math 483/583 - High Performance Scientific Computing

Instructor
Chris Swierczewski

Teaching Assistants
Meghana Velegar - 583
Sean Santos - 483

Overview

- "High Performance Scientific Computing"
- Course goals and syllabus
- Homework, quizzes, and final exam
- This week's homework
- Demo of SageMathCloud

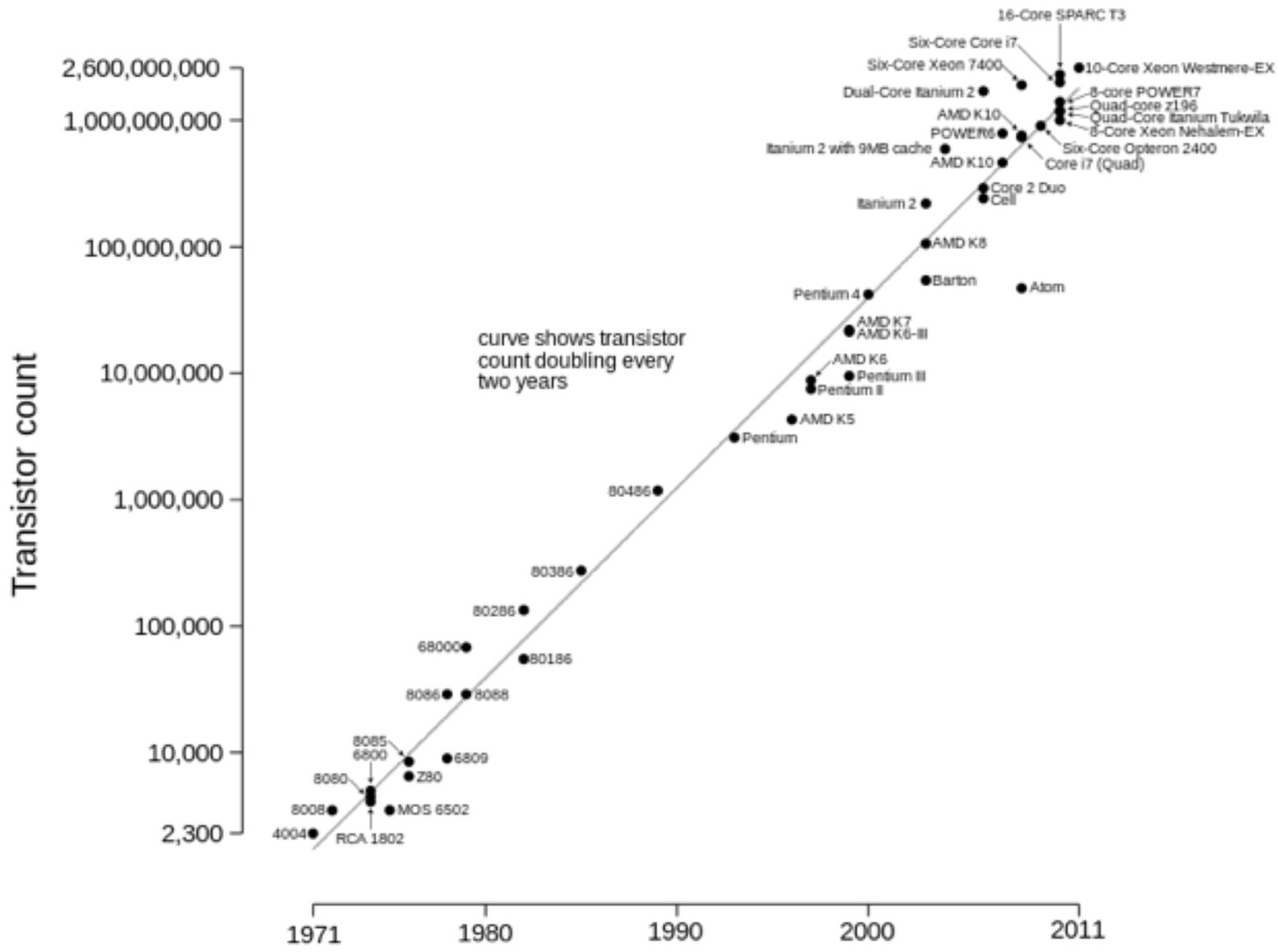
Course Sections

	In-Class	Online	Online Masters / EDGE
AMath 483	A SLN 10212	B SLN 21245	
AMath 583	A SLN 10230	C SLN 21246	B SLN 10231

High Performance Scientific Computing

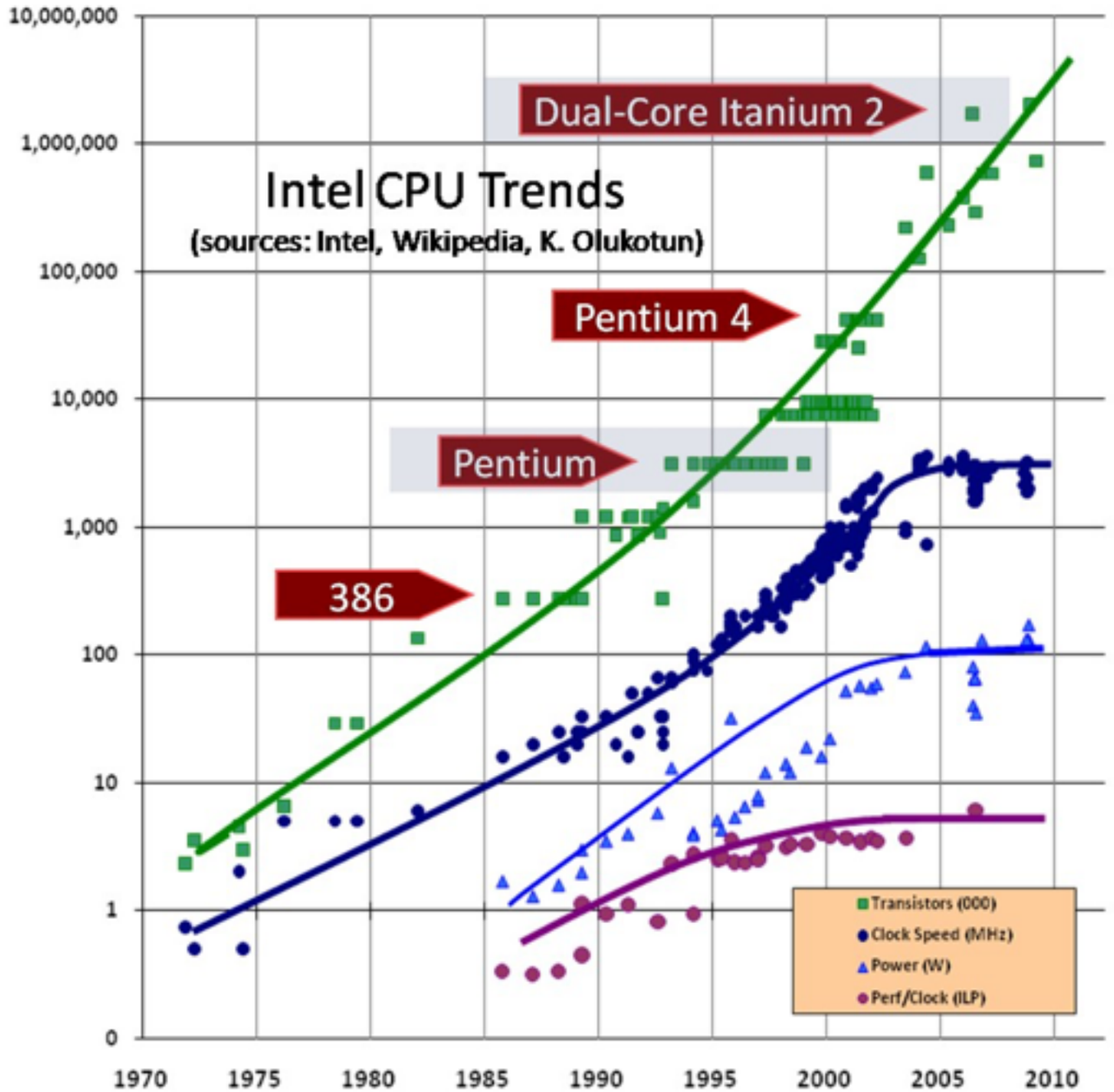
- “High Performance” - multiple cores, computers, or clusters with hundreds to thousands of cores

- “Scientific Computing” - problems that appear in optimization, numerical analysis, geometry, data analysis



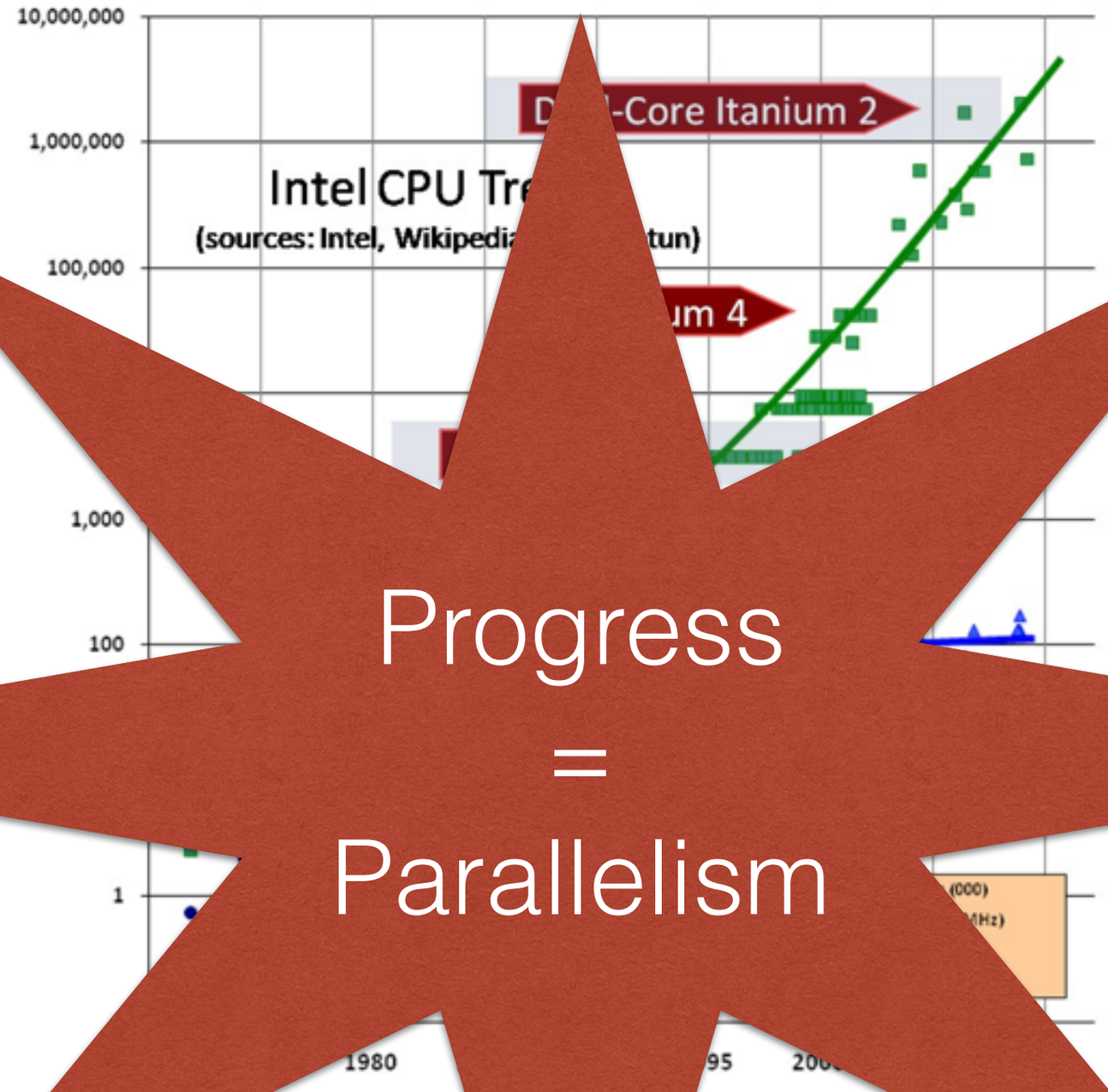
Moore’s Law

transistors on a CPU doubles every two years



Moore’s Law

*but clock speed and FLOPS are not doubling every two years



Progress

=

Parallelism

Moore's Law

*but clock speed and FLOPS are not doubling every two years

MILKYWAY-2

Current world’s fastest supercomputer.
http://top500.org

Course Goals

- techniques for writing fast single-core, multi-core, and multi-machine code
- tools for managing “programmer time”
- applications and tools for scientific computing

Knowledge gained from course will be useful for both supercomputers as well desktop environments.

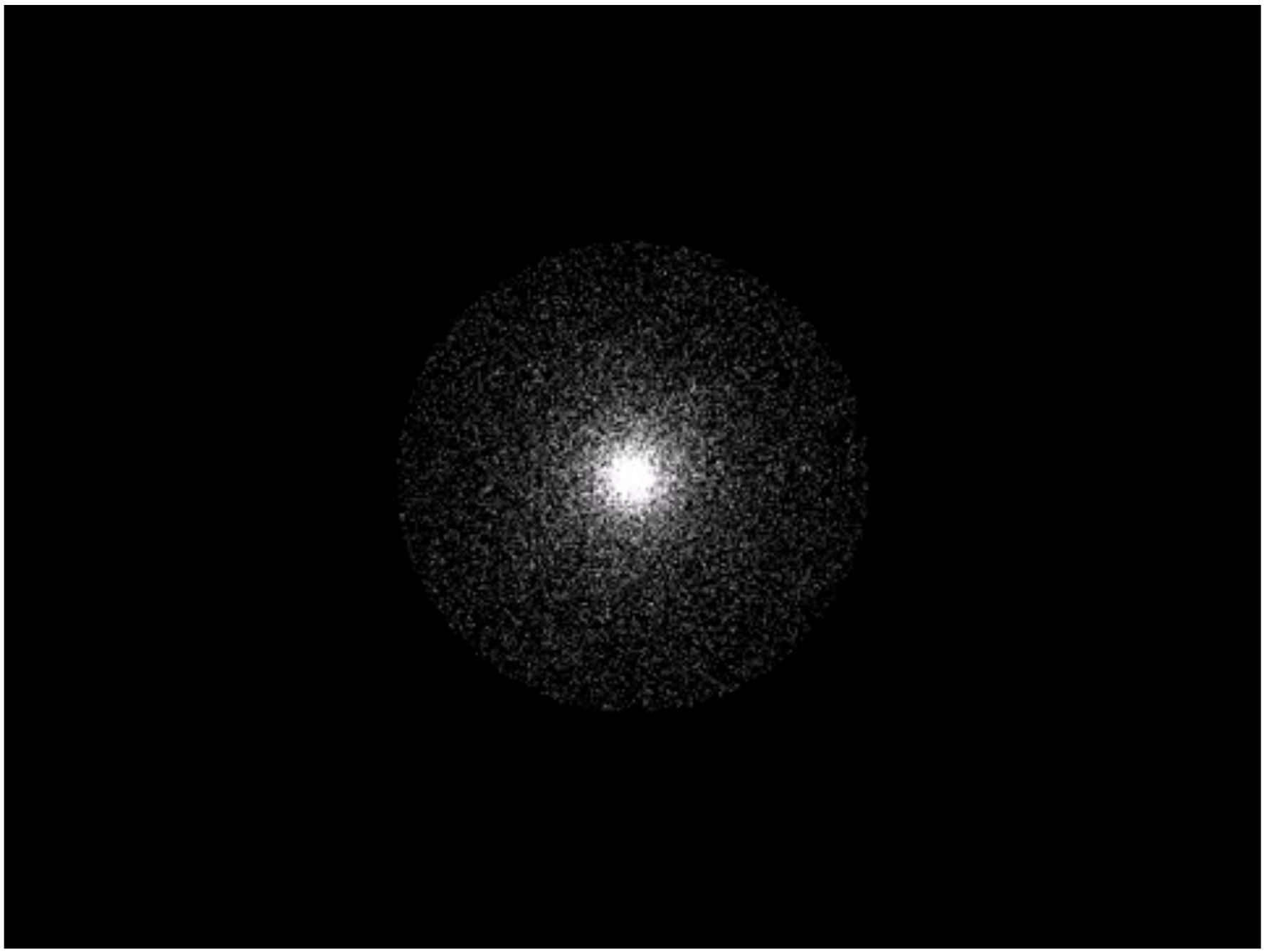
Course Goals

- **Software efficiency**
  - computer hardware: CPU, memory, floating point arithmetic, cache
  - Unix and command-line software development
  - languages: Python and C
  - efficient single-core programming
  - parallel programming with OpenMP and MPI

Course Goals

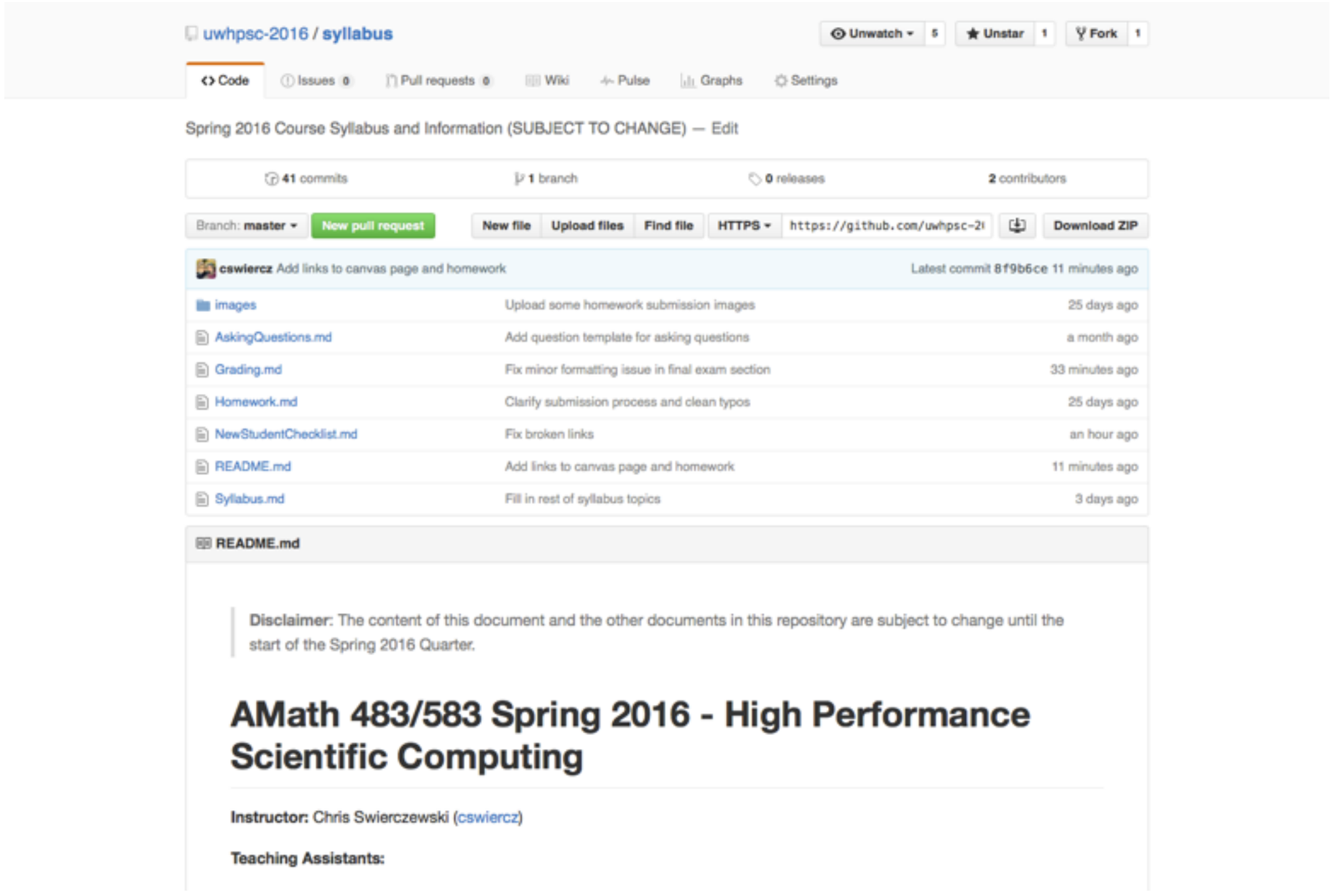
- **Programmer efficiency**
  - version control systems
  - makefiles and scripting
  - debugging code
  - test suites
  - reproducibility: documentation and reusability

N-Body Simulation



Parameter Optimization

Syllabus (link)



Grading

40%	Homework x4
40%	Quizzes x8
20%	Final Project + Exam

Homework - 40%

- Automated Tests
  - “toy” test suite given to students
  - additional tests described, but not given
- Report and Analysis
  - derivations, analysis, plots

60 / 100	Automated Tests
20 / 100	Report and Analysis
10 / 100	Documentation and Reproducibility
10 / 100	Performance*

Homework - 40%

- Documentation and Reproducibility
- Performance
  - - 3 std. dev. — 0 / 10 pts
  - - 2 std. dev. — 5 / 10 pts
  - +/- 1 std. dev. — 10 / 10 pts
  - + 2 std. dev. — 15 / 10 pts

60 / 100	Automated Tests
20 / 100	Report and Analysis
10 / 100	Documentation and Reproducibility
10 / 100	Performance*

Quizzes

- Every Monday (including next week!)
- Understanding of Lecture Content and Assigned Reading from previous week
- Taken online via Canvas
  - 15 minutes timed
  - Start: Monday at 4:15pm
  - End: Monday at 4:45pm

Academic Integrity

- **Dont's:**

  - you may not work as a partner with another student on an assignment,

  - you may not show your solutions to another student or look at their solution,

  - you may not discuss homework solutions in the course chat room or on the Issues pages,

Academic Integrity

- **Dos':**
  - asking for help in understanding a part of a lecture,
  - assistance in the use of a software tool or library
  - requesting a reference on a given software tool or library,
  - discussion of topics beyond the scope of this course. (It's fun!)

Asking Questions

- I will not answer email about the course after the first week.
- All questions should be posted to the appropriate repository's Issues Page in GitHub
  - logistics questions —> Syllabus Issues
  - Homework #1 Questions —> Homework #1 Issues Page
- Gitter Chat Room
  - online office hours (TBD)

Lecture Notes

- See “Lectures” repository
  - slides
  - code
  - other notes

Philosophy of the Course

- Many topics to cover! - Not enough time.
- Lectures are “introductions” to topics.
- Lectures and Homework give enough “hands on” experience for further experimentation.
- **Primary** and **Secondary References** (see Syllabus) are major source of information.

Philosophy of this Course

- There is no textbook for life, but there are countless resources.

- Learn how to find information on the internet, in books, from each other.

Prerequisites

- Courses:
  - Math 301 or CSE 142
- Highly recommended
  - solid linear algebra background
  - solid programming skills

Before we continue…

- **Homework** - follow instructions on the **New Student Checklist**

- **Homework** - Learn Python

  - quiz next Monday - basic Python, Unix, and Git

  - **Official Python Tutorial** Sections 3,4,5,6,14

Demo: SageMathCloud and the Terminal