CS2204

Program Design and Data Structures for Scientific Computing

Spring 2016
MWF 10:10-11:00pm
FGH 258

Course Staff: Instructor

- Dr. Graham Hemingway
- graham.hemingway@vanderbilt.edu
- Cell Phone (615) 294-7133
- Office Hours (in FGH Atrium)
  - M/W 11:00 - 12:00
  - Tues 3:00-4:00
  - Or any time by appointment

Course staff: TA’s

- TA/Grader’s, their email addresses, and their office hours will be posted on Oak
  - TBD

- TA’s will hold office hours in the EECS computer lab, FGH 201, starting next Wednesday.
  - Look for the person with the “**CS2204 TA**” placard
  - Please let me know if a TA does not show up for their office hours
  - TA’s for other courses are not required to help you (and likely will not)

Course Objectives

- Data Structures
  - Learn different ways to store collections of data in a computer program
  - Which data structure is preferred in a particular situation

- Program Design and Implementation
  - Object-oriented programming
  - Abstract data types
  - Algorithm design & analysis (time & space complexity)
  - Various problem-solving strategies

- Programming skills
  - Advance your skills & abilities as a programmer
  - How to write efficient and maintainable programs
  - Use predefined libraries of functions

Course Overview

- Course syllabus
- Blackboard/OAK
- 3 exams
- Comprehensive final
- As many programming assignments as we can fit in (hopefully a lot!!)

Prerequisite

- CS1101 or CS1103 (or equivalent) is the prerequisite for CS2204
- It is assumed that you have studied Java, Matlab, or C++
- Topics you are expected to have mastered:
  - An intro-level ability of problem solving and programming
    - Problem decomposition, design, coding, testing
  - Variables, data types, expressions & assignment
  - Various control flow constructs
  - Program input & output
  - Methods/functions & parameter passing
  - Arrays & array processing

Prerequisite

- CS2204 is a terminal CS course
  - Its main purpose is to support the Scientific Computing minor

- CS2204 is not a sufficient prerequisite/preparation for any other computer science courses
  - If you think you might want to take other CS courses in the future, you should be in CS2201 rather than CS2204
  - CS2201 is a prerequisite for almost all CS classes and also satisfies the requirements for the Scientific Computing minor
  - After completing CS2204 you will be prepared to take SC2250 (note that is SC2250 and not CS2250)

Text book

- Problem Solving with Algorithms and Data Structures using Python - Second Edition
  - Second edition of the book uses Python 3.X
  - Second edition available online for free!

No Cheating!

I am serious!

To avoid any problems, read & understand the course's Academic Honesty Policy posted on Oak (under Course Documents).

Note: if you already know some of the material in this class, resist the temptation to help your struggling friends too much. Providing code to a friend is as bad receiving it – neither are allowed. This is a source of many honor code violations

- Also do not share computers or passwords with classmates.
- Do not copy code from past students of the class.

Programming Projects

- "Homework assignments are about building knowledge through practice. It's about confronting the assignment, thinking about it, working through some blind alleys, and eventually coming up with a solution. Parsing out the true goal of the assignment, it's not about solving the assignment. It's about the process. The world does not need another solution to the Random Sentence Generator or Tetris or whatever. It's like running laps around the track. It's not about getting somewhere. It's about running."
    - Nick Parlante

- The goal of the programming projects is not simply to turn in a working program, but it is learning how to *create* a working program.
    - Cheating robs you of this learning experience.

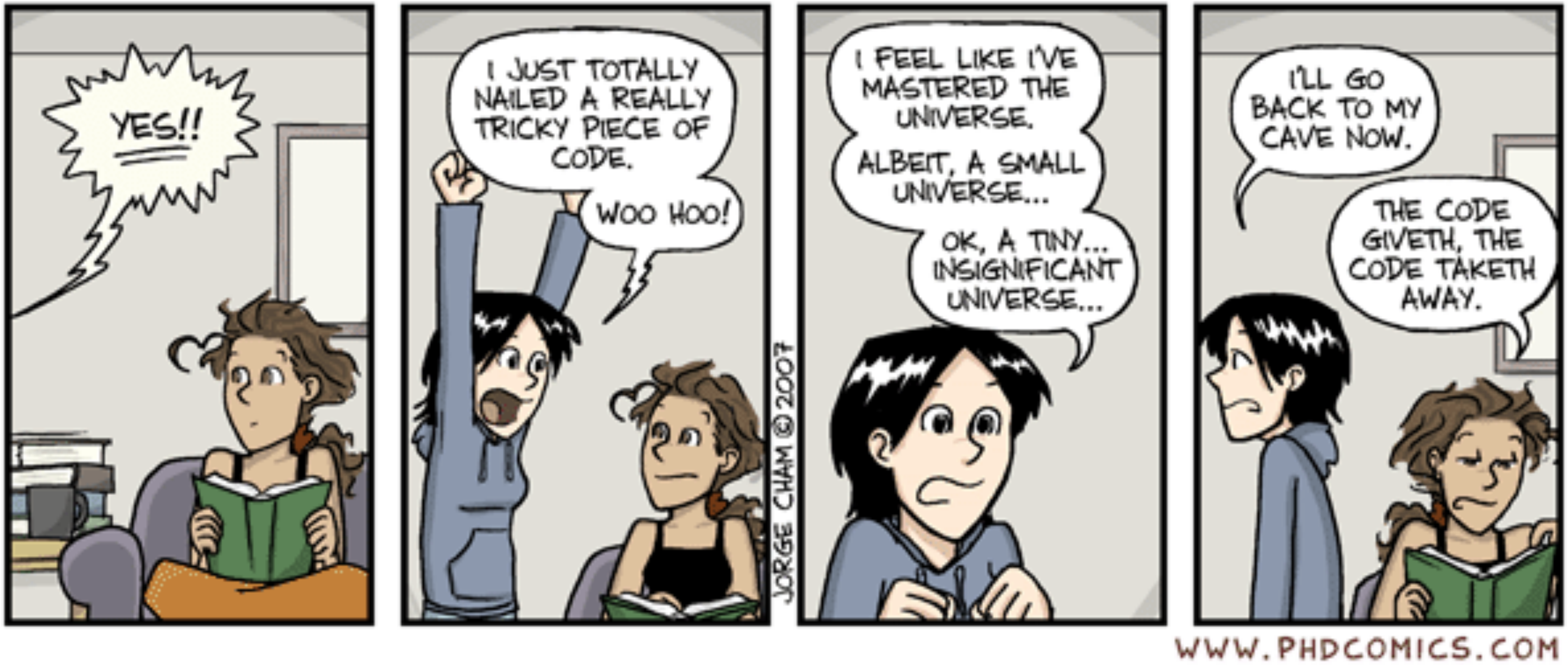
Python and PyCharm

- We will be using the Python programming language
  - We will utilize the latest Python 3.x
- We will be using the Anaconda version of Python
  - Freely available online
  - Contains all the needed modules (NumPy, SciPy, etc.)
- We will use the PyCharm IDE for program development
  - You can use an alternate IDE if you want
- All the software we will use is available for **free!**
  - Download instructions are available on Oak

Hints for success

- Attend lecture (and pay attention!)
  - I plan to lecture on the whiteboard in this class, so I may not be posting lecture notes to Oak
  - Ask questions
  - Meet with your friends to review/discuss lecture material
    - I am exploring the use of OAK (or other tool) to host online discussions
- Complete any reading assignments
- Understand programming assignments before attempting to code them
- Develop & test your programs incrementally
- Use your brain rather then the compiler & debugger

A Sense of Accomplishment


YES!!
I JUST TOTALLY NAILED A REALLY TRICKY PIECE OF CODE.
WOO HOO!
JORGE CHAM © 2007
I FEEL LIKE I'VE MASTERED THE UNIVERSE.
ALBEIT, A SMALL UNIVERSE...
OK, A TINY... INSIGNIFICANT UNIVERSE...
I'LL GO BACK TO MY CAVE NOW.
THE CODE GIVETH, THE CODE TAKETH AWAY.
WWW.PHDCOMICS.COM

Announcements

- Programming assignments will hopefully start next week
- Make sure you have Python and PyCharm installed (or some alternate Python programming environment)
  - We will do a demonstration in lecture next class
- Please turn in your student survey at the end of class