# Numerical Methods in Materials Science and Engineering

## First Day

Matthew Goodman
mgoodman@email.arizona.edu

Materials Science and Engineering
University of Arizona

August 24, 2009

# Welcome to MSE 350

- I am not Dr. Erdmann! He is **out** for the week.
- Watch the NASA launch (`http://countdown.ksc.nasa.gov/shuttle/countdown/cdt/`)
- Laptop use in class is encouraged! (for programming related . . . )
- There is no assumption of previous programming experience . . .
- Prereqs: Vector and Diff-Eq.

# Today

- This presentation:
  `http://u.arizona.edu/~mgoodman/`
- Class site:
  `http://www.u.arizona.edu/~erdmann/mse350/`
- Plagiarism Video
  `http://deanofstudents.arizona.edu/`
- Class overview
- HW for Wednesday

# Academic Integrity "video"

## Just don't do it!

**Fwd: New Academic Integrity video to show students on the first day of classes** Inbox | X

**Robert G. Erdmann** to Ge( show details Aug 20 (2 days ago) Reply

Hey,

I just got this. As directed below, please show the academic integrity video during the week I'm gone. Also please emphasize that if the students cheat, I will personally hunt them down and brutally murder them regardless of the university policy that likely prohibits this.

Thanks.

-RGE

# What is Python?

Python is a general-purpose high-level programming language. Its design philosophy emphasizes code readability. Python claims to "[combine] remarkable power with very clear syntax", and its standard library is large and comprehensive. Its use of indentation as block delimiters is unusual among popular programming languages.[wik, ]

# Why Python?

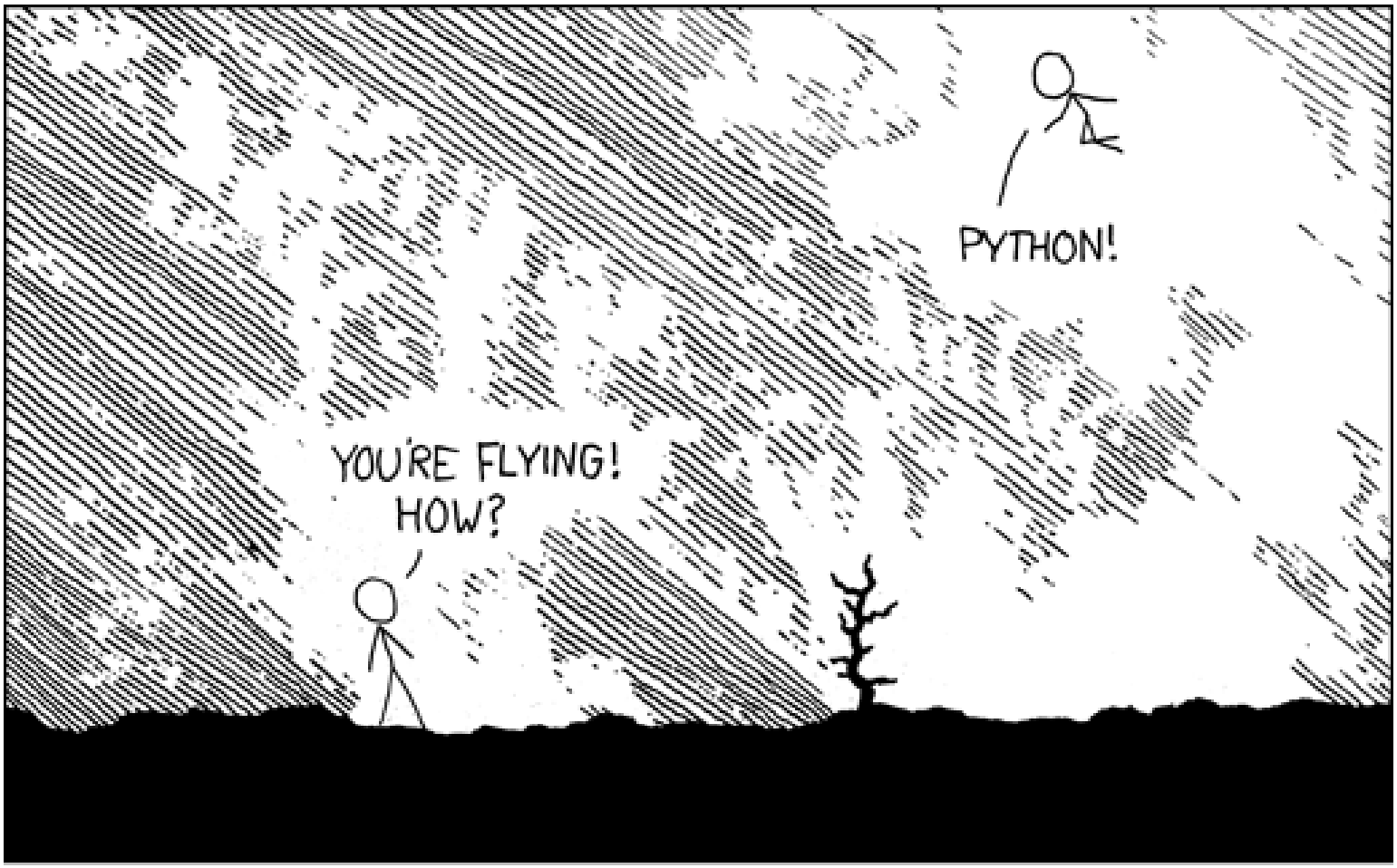
- Modern
- Modular
- Concise
- Powerful
- Interactive
- Cross Platform

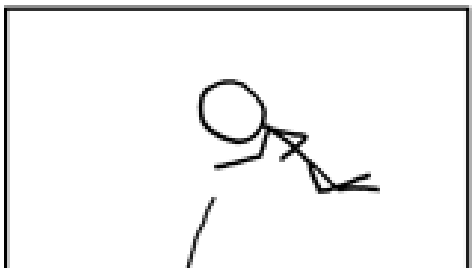
# Why Python? – Modern

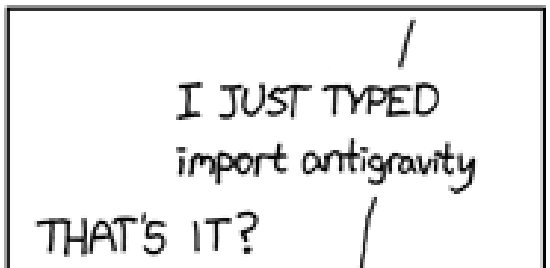
- First Release in 1991!
- Under heavy active development
- Used by a number of giants:
- YouTube, Google, BitTorrent
- Yahoo!, CERN, NASA, OLPC
- Most Linux/Unix

# Why Python? – Modern

## From xkcd.com[xkc, ]

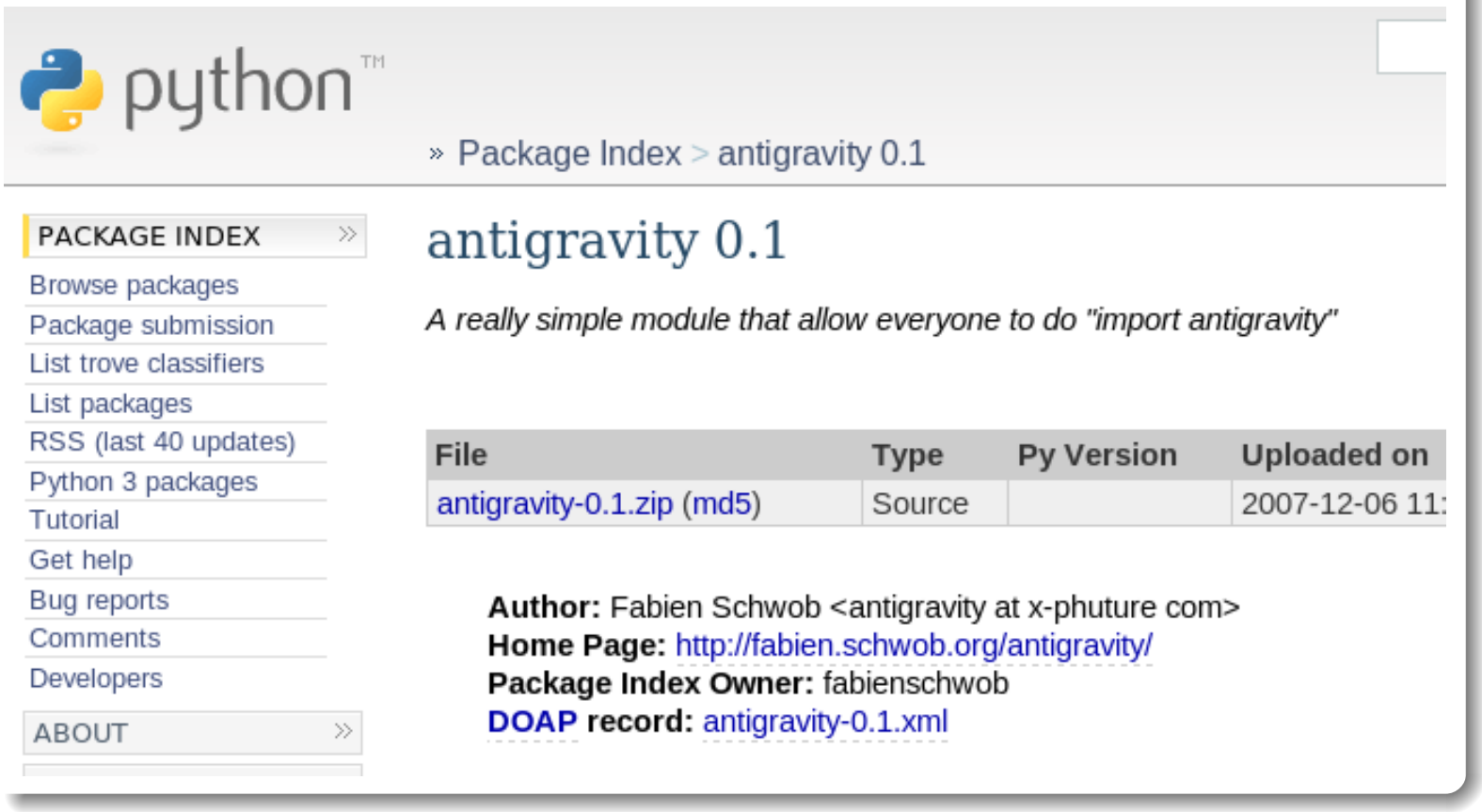


# Why Python? – Modern

## Wow Development!



# Why Python? – Modular

Some standard modules:

- os – Dealing with the operating system and files
- wave – Working with sound files
- gzip, tarfile, zipfile – Working with compressed files
- urllib, webbrowser – Retrieving web pages and interacting with the local web browser
- sqllite, tables, csv – Specialized database and data file formats.
- email, pidgin – Interface to email and chat clients.

# Why Python? – Modular

Unique problems in programming are RARE.

- Reinventing the wheel == bad!
- Python modules offer engineered solutions to common tasks.
- This class will focus on several modules well suited to engineering and science related endeavors:
  - Numpy – Powerful array operations
  - Scipy – Scientific libraries
  - MatPlotLib – Plotting `http://matplotlib.sourceforge.net/gallery.html`
  - Mayavi – 3d visualization (demo) `https://svn.enthought.com/enthought/wiki/Mayavi/Gallery`

# Why Python? – Modular

Modularity extends even to OTHER PROGRAMMING LANGUAGES

- SWIG – Generates python interfaces to many languages (Java, C, C++, FORTRAN, Perl . . .)
- Cython – Creates C code out of Python
- PyCUDA – Interface to the NVIDIA CUDA GPU computing platform

# Why Python? – Concise

- Python was designed with readability in mind.
- Code is broken up with white-space
- More on this all later!

print “hello world”
lunch_options = “spam, eggs, and spam”
if “spam” in lunch_options: print “Yay spam!”

# Why Python? – Powerful

All of the following things could be **comfortably** accomplished in less than 10 lines of code thanks to the wealth of modules available:

- Monitor an industrial process and send an email in case of disruption.
- Connect to a remote database and perform a complex query and display/save the results.
- Take the symbolic integral of a complex function.
- Visualize a 2d or 3d data set with interactive display
- Download a large number of files from a website on a automatic schedule.
- Load an image, perform several filters and save it as a different format.
- Write an alarm clock with user specified wait time and music ramping.
- Make a chat-robot that harasses your friends.

# Why Python? – Interactive

Demo time!

- Interactive help
- Interactive programming
- Interactive HPC

# Why Python? – Cross Platform

- Supports all major operating systems
  - Linux
  - Windows (95, XP, Vista, 7?)
  - Mac OSX
- Portable code!

# HW1: Get Python

Downloading and Installing the following is your homework for the next class:

- python – the Python interpreter
- ipython – the interactive Python interpreter
- A text editor – 100+ options here . . .

# HW1: Continued . . .

Also, we will need the following modules:

- numpy
- scipy
- matplotlib
- tables
- mayavi

# HW1: Made Easy

You could track down all these individual installs and files, but it is much easier to download a bulk distribution that includes all of these tools:

- The Enthought Python Distribution (EPD) found at `http://www.enthought.com/`. (all platforms)
- python(x,y) the "full" distribution found at `http://www.pythonxy.com/`. (ia-64 not supported?)
- From the Linux apt repositories (ask me for a package list) (Debian/Ubuntu)

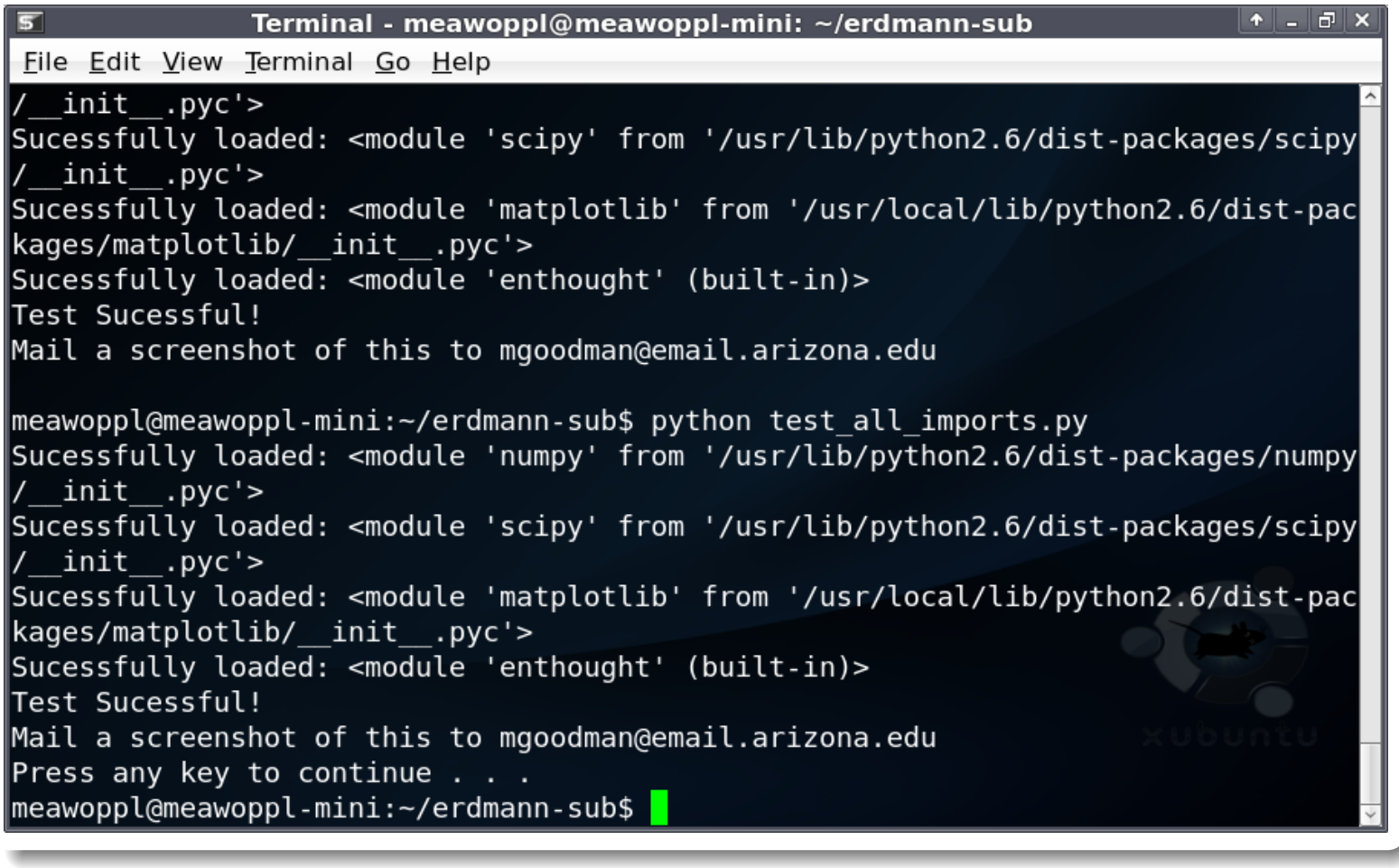
# Completing HW1

The final three steps:

1. Download the python script found here (`http://www.u.arizona.edu/~mgoodman/`)
2. Run it
3. Send a screen shot of everything working well to me at mgoodman@email.arizona.edu
4. Please include your full name, and a "secret-name/moniker" under which grades will be posted (if you don't choose one, I will and you probably wont like it)

# Working Configuration

## This is good:



# Lastly . . .

- If you have any trouble . . .
- Please also begin reading the first three sections of the python tutorial found at:
- `http://docs.python.org/tutorial/`
- We will have covered all of that material by the end of Wednesday if all goes well.

# Bibliography I


Wikipedia, the free encyclopedia.
`http://www.wikipedia.org`.

Xkcd, a webcomic of romance, sarcasm, math, and language.
`http://www.xkcd.com`.