Plan For The Day (PFTD)
- Practice solving problems (algorithms, programs)
  - Some solved with a computer, some with Python
- Learning about vocabulary
  - We'll work with English and Python

- Practice using tools for Duke Compsci courses
  - Eclipse, APT, ambient, ... Python-tutor
  - Sakai, Piazza, Feedback

- Reveling in the wonder of thinking and working
  - How do we know when something works?

Who took Compsci 101?

Why is programming fun?  Fred Brooks
- First is the sheer joy of making things
- Second is the pleasure of making things that are useful
- Third is the fascination of fashioning complex puzzle-like objects of interlocking moving parts
- Fourth is the joy of always learning
- Finally, there is the delight of working in such a tractable medium. The programmer, like the poet, works only slightly removed from pure thought-stuff.

Algorithm
- Recipe
- Sequence of steps that constitute instructions
- Step-by-step procedure for calculations

What does Nate Silver do?  
http://53eig.ht/1tZy909

How do Netflix and Amazon know me?  
compsci101: capable of implementation as a program, but tread gently here
http://moreintelligentlife.com/content/features/anonymous/slaves-algorithm
Skills and Practice for Game Playing

- I have 7 and 5, dealer showing 5, I should ...

Skills and Practice for Game Playing

- http://www.youtube.com/watch?v=AEBbsZK39es

$193, $540, $820, $700, $749. Are these reasonable? Why?

I'm thinking of a number ...

- You guess. I'll tell you high, low, or correct
  - Goal: guess quickly, minimal number of guesses
  - Number between 1 and 100...
  - Number between 1 and 1000...

- Can you describe an algorithm, instructions, that would allow someone to use your instructions to play this game correctly. Start with 1 and 100, but ideally your instructions work with 1 and X


Analyzing the binary search algorithm

- Is the algorithm correct?
  - Try it, again, and again and ...
  - Reason about it: logically, informally, ...

- How efficient is the algorithm?
  - How many guesses will it take (roughly, exactly)
  - Should we care about efficiency?

- When do we really care about efficiency?
  - Examples?
Concepts you'll learn in Compsci 101

- **Programming**
  - Practice, skill, art, science, engineering, creativity
- **Problem-solving**
  - How to solve problems using programming and a computer
- **Impact of computer science**
  - Scale and automation: powerful forces
- **Foundation for future work**
  - In many areas, not limited to compsci@duke

Programming Examples

- **Scratch example from class, elaborate in lab**
- **Hour of code**: [http://learn.code.org/hoc/1](http://learn.code.org/hoc/1)
  - Designed for kids, useful to millions
- **Light-bot** (many versions, first assignment)
- **Python!**

Python and Programming Concepts

- **Names are important, abstractions**
  - What is [http://152.3.140.1](http://152.3.140.1)
  - What is [http://www.amazon.com](http://www.amazon.com)
- **Types are important, facilitate operations**
  - What is foo.pdf, foo.mp4, foo.jpg, foo.wav
  - Do the file extensions guarantee file type?
- **Thinking in terms of names and types can help**
  - Python has types, inferred dynamically
  - Python uses types differently from Java and C++

Latanya Sweeney

I am a computer scientist with a long history of weaving technology and policy together to remove stakeholder barriers to technology adoption. My focus is on "computational policy" and I term myself a "computer (cross) policy" scientist. I have enjoyed success at creating technology that weaves with policy to resolve real-world technology-privacy clashes.

Identify 87% of US population using (dob,zip,gender). Director of Harvard Data Privacy Lab, instrumental in HIPAA because if de-identification work, currently Chief Technologist FTC
(RE)-INTRODUCTION TO PYTHON

Python data reading code

```python
f = open("/data/kjv10.txt")
st = f.read()
len(st)
ac = st.count('a')
zc = st.count('z')
for ch in 'aeiou':
    print ch, st.count(ch)
```

Vocabulary, grammar, rules: Python

- **Naming**
  - The power of abstraction and parameterization
  - What is abstraction?
  - What are parameters? What has them?
- **Types**
  - What's used in Python? Use console
  - Determine names of types in Python
  - int, float, bool, string, list, ...
  - Operators and expressions: (see web pages)

Variables, Types, Values

- **Variable is a name associated with "container/stuff"**
  - Assign a value: `x = 5`
  - Print value of variable: `print x`
  - Use variable in expression: `y = x * 55`
- **String is a type and has a value**
  - Assign: `x = "hello"`
  - Print value of variable: `print x`
  - Use in expression: `x + " world"`
Expressions, Operators, Types

- Why is $3+5\times4$ different than $(3+5)\times4$?
  - Operator precedence: $()$ are your friends

- Why is $5/3$ different than $5.0/3$?
  - We use Python 2.7, different in Python 3.0

- What happens when operators go bad?
  - What is "apple" + 3? What is "apple" + "pi"?
  - What is "apple" * 3? What is "apple" * "pi"?

ALGORITHMIC INTERLUDE

Whole Genome Shotgun with words

- Creation algorithm
  - Take a phrase
  - Replicate it four times
  - Chop into "chunks"
    - 15-22 characters

- How to recreate original phrase?

Getting ready to code in Python

- We need a programming environment
  - Eclipse, PyDev, Python, Ambient
    - Open source or free for academics

- We need a computer with an operating system
  - Installing the suite of tools can be cumbersome
    - Persist, Persever, Get Help, start over

- Getting used to the environment can take time
  - Once you've got it, second nature!
    - Easy to reuse with a new language
What is an APT? **BMI APT**

- **Automated/Algorithmic Problem Testing**
  - Write one function, 2-30 lines, solve a problem
  - Tested automagically in Eclipse or the browser
  - Test test test ... Quality of code not an issue

- **Start simple, build toward more complex**
  - What is a function? A function call?
  - What is a parameter? Argument?
  - How do you run/execute a program