Objectives

- Introduce you to new ideas in computation and information that will make you a better problem solver.
- Teach you how to use the Python language.
- Familiarize you with programming ideas and techniques that transcend languages.
Class Structure

- Three 50 minute interactive lectures on MWF at 10 am
- Two 90 minute recitation/lab sections on TW (attendance at only one is required)
- Class work, weekly homework, three tests
- Interweaving of mathematics, computation and thought
- Problem-oriented learning: as far as possible I will choose real-world problems from a variety of fields of study to stimulate learning
- All information posted on Owlspace as well as at http://www.bandgap.cs.rice.edu/classes/comp140
Background

- A distribution course open to any one in engineering, natural sciences or social science.
- High school level mathematics assumed.
- No programming experience assumed.
- A curious mind is a plus.
Course Staff

- Devika Subramanian
- Stephen Wong
- Emily Fortuna
- Allen Gregory
- Nick Hirsch
- James Russell
- Drew Bryant
- Beth Crompton
The devil and Daniel Webster

The devil made a proposition to Daniel Webster. The devil proposed paying Daniel for services in the following way: "On the first day, I will pay you $1,000 early in the morning. At the end of the day, you must pay me a commission of $100. At the end of the day, we will both determine your next day's salary and my commission. I will double what you have earned at the end of the day, but you must double the amount that you pay me. Will you work for me for a month?"
Computational thinking (CT)

- “Word problems”
  - *Goal of computation*: What are we trying to decide on/determine?
  - *Information extraction*: what relevant information has been provided?
  - *Algorithm/recipe design*: how do we calculate answer from givens?
  - *Algorithm/recipe implementation*: how do we get a computer to “cook” your recipe?
CT in everyday life

- Looking up a word in a dictionary/name in a telephone book
  - Google’s index of Web documents
- Deciding on which line to stand in a bank, supermarket, customs & immigration
  - Queuing theory/task scheduling
- Packing items in your backpack
  - Knapsack problem
- Taking ones kids to soccer, gymnastics, and swim practice
  - Traveling salesman problem
CT in everyday life

- Cooking a gourmet meal
  - Task decomposition/parallel processing
- Cleaning out your garage
  - Paging algorithms (managing the memory hierarchy)
- Storing away your Lego pieces scattered on the floor/your books/your CD collection
  - Indexing and hashing
- Even in grade school, we learn algorithms (long division, factoring, GCD, ...) and abstract data types (sets, tables, ...).
Challenges of scaling up

- The number of potential solutions is so large that even a fast computer cannot evaluate all these solutions.
- Some solutions are cheap, and others may be expensive or undesirable.
- How can we find these good solutions?
- CT tells us how!
Who uses CT?

- Who decides routes for trash collection? (GeoRoute)
- Who designs routes for your mailman/UPS/FedEx? (RouteSmart)
- http://www.time.com/time/magazine/article/0,9171,1814175,00.html
Finding gas prices online
What is CT?

- Is conceptualizing, not programming
  - Computer science IS NOT computer programming
- Is a fundamental, not a rote skill
- Is a way humans, not computers think!
  - Humans are clever and creative
  - Computers are dull and boring
- Complements and combines mathematical and engineering thinking
Modern computer science

- Formulating fuzzily defined problems
  - Finding the right level at which to model the problem
  - Extracting the pieces that are interesting and are solvable

Devising new randomized strategies that work with high probability

Focus on solution quality, not on running time.
What can you do with a CS degree?

- ANYTHING YOU WANT TO DO!
  - Medicine
  - Law (patent law)
  - Grad School --> professor
  - Drug designer
  - Wall Street maven
  - Computational Artist
  - Search Technologist
  - Animator/Game designer
  - ... and many many more
Thinking outside the box

"I know it's unconventional, but there's only so much homework one dog can eat."

http://www.andertoons.com/cartoon/2705/
Sources

- Jeannette Wing essay and talk on *Computational Thinking*
- S. Khuller talk on algorithms in everyday life