

Objectives

- Computer Science is Complexity Science

Apr 8, 2016

Sprengle - CSC1111

1

Review

- What are common constructs in programming languages?
- What are some differences between programming languages?

Apr 8, 2016

Sprengle - CSC1111

2

Computers are incredibly fast, accurate, and stupid.
Human beings are incredibly slow, inaccurate, and brilliant.
Together they are powerful beyond imagination.
-- Albert Einstein

Apr 8, 2016

Sprengle - CSC1111

3

COMPLEXITY SCIENCE

Apr 8, 2016

Sprengle - CSC1111

4

CS == Complexity Science

- How can it be done?
 - Based on **information**
 - Managing, manipulating data
 - Possible algorithms
- How well can it be done?
 - Most **efficient** algorithm in terms of time and/or space
- Can it be done at all?
 - Often, proof is a program--an implementation of the above

Apr 8, 2016

Sprengle - CSC1111

5

Computer Science != Programming

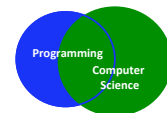
programming : CS ::

machining : engineering

grammar : literature

equations : mathematics

walking : W&L



a vehicle, not a destination

Apr 8, 2016

Sprengle - CSC1111

6

Computer Science Fields

Systems

- Architecture
- Operating systems
- Networks
- Distributed and parallel systems
- Databases
- Security
- ...

Software

- Compilers
- Graphics
- Software engineering
- Software testing and verification
- ...

Theory

- Algorithms
- Theory of computation
- ...

Other

- Artificial intelligence
- Robotics
- Natural language processing
- Bioinformatics
- Visualization
- Numerical analysis
- ...

- Often research involves combinations of these fields
- Not just programming!
 - But programming is a tool to do much, much more!

Apr 8, 2016

Sprengle - CSCI111

7

Computer Science Fields

Systems

- Architecture *
- Operating systems *
- Networks *
- Distributed * and parallel systems
- Databases *
- Security*
- ...

Software

- Compilers
- Graphics *
- Software engineering *
- Software testing * and verification
- ...

Theory

- Algorithms *
- Theory of computation
- ...

Other

- Artificial intelligence *
- Robotics *
- Natural language processing
- Bioinformatics
- Visualization*
- Numerical analysis
- ...

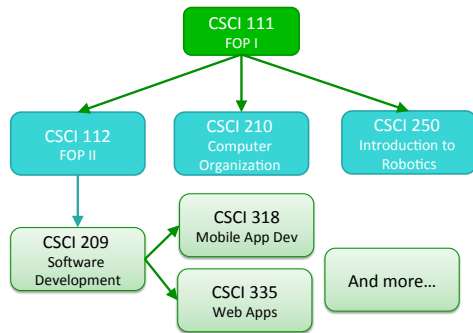
- * = field we discussed or did a problem in
- Some are a stretch :)

Apr 8, 2016

Sprengle - CSCI111

8

Where Can You Go from Here?



Apr 8, 2016

Sprengle - CSCI111

9

Conclusions

- See impact of computer science on your life
 - Think differently about issues
- Understand some of the computing issues better
 - Taking out some of the mystery
 - Testing, debugging, efficiency
- Algorithms are everywhere
 - Process for solving problems, **efficiently**
 - Mapping human intuition to systematic/automatic process

Apr 8, 2016

Sprengle - CSCI111

10

Final Exam Take Home Questions

- 2 essay questions about the Broader Issues
- **Due before end of exam period**
 - 5 p.m. Friday
- Each essay should be about 1/2 a page, single-spaced
- Goal: answer the question clearly, precisely, and convincingly
 - Not too wordy
 - Evidence/examples to support your argument
 - Correct spelling, grammar, punctuation

Apr 8, 2016

Sprengle - CSCI111

11

Final Exam

- Finals are taken in the lab classroom (Parmly 405)
 - No computers
 - If need to change your time, sheet outside the CS department office
- Evaluations due Sunday at midnight on Sakai
- Take-home essay due Friday at 5 p.m.
 - End of exam period
- All lab work and extra credit articles must be submitted by **Wed**
- Office hours:
 - Monday: 10 a.m. – 5 p.m.
 - Tuesday: 9:10 a.m. – 5 p.m.
 - Thursday: 9:10 a.m. – 2:30 p.m.
 - Appointments preferable during that time
 - Others by appointment
 - Can email about other appointments as necessary

Apr 8, 2016

Sprengle - CSCI111

12

Final Exam Review

- Focus on object-oriented programming
- New content: dictionaries, search techniques, lists (1D and 2D), complexity science
- Cumulative:
 - Functions, data types, common methods & operations
 - How to model data

Your questions?

Apr 8, 2016

Sprengle - CSCI111

13

Final Exam Review

- What do you need to do to be able to use methods from a class?
- What are the different ways to iterate through a list?
- How can you iterate through a dictionary?

Apr 8, 2016

Sprengle - CSCI111

14

Exam 2: Analysis of Code

```
def helper1(word, letter):  
    for i in range(len(word)):  
        if word[i] == letter:  
            return i  
    return -1
```

```
def helper2(word, letter):  
    total = 0  
    for ch in word:  
        if ch == letter:  
            total += 1  
    return total
```

Apr 8, 2016

Sprengle - CSCI111

15

Counter Class Specification

- Implement, Test
- Example use: Caesar cipher

- A class that represents a counter that wraps around from a high value back to its low value
- Functionality:
 - Constructor – takes as parameters the low value and the high value; default – counter starts at low value
 - A string representation of the Counter
 - Increment the counter by a given amount (a positive amount), wrapping around to low again, if necessary. Returns number of times had to wrap around.
 - Decrement the counter by a given amount (a positive number), wrapping around to high again, if necessary. Returns number of times had to wrap around.
 - Sets the counter's value, only if low <= value <= high. Otherwise, prints an error message.
 - Getters: low, high, current value

Apr 8, 2016

Sprengle - CSCI111

16

Course Evaluations

- On Sakai, due Sunday
- Incentive
 - If 60% of students complete evaluation, 1% Extra Credit on lab grades
 - For each additional 10% of students who complete evaluation, 1% EC on lab grades
 - Total possible EC: 5%

Apr 8, 2016

Sprengle - CSCI111

17