Implementing CS Principles as a Breadth-First Survey Course

Chris Mayfield <mayfiecs@jmu.edu>

Department of Computer Science, James Madison University

About the Course

Traditional Survey Course for CS Majors + **CS Principles Curriculum Framework** = **JMU CS 101**

	Creativity						Abstraction						Data						Algorithms						Programming						Internet				Impact						
	LO 1.1.1	LO 1.2.1	LO 1.2.2	LO 1.2.3	LO 1.2.4	LO 1.2.5	LO 1.3.1	LO 2.1.1	LO 2.1.2	LO 2.2.1	LO 2.2.2	LO 2.2.3	LO 2.3.1	LO 2.3.2	LO 3.1.1	LO 3.1.2	LO 3.1.3	LO 3.2.1	LO 3.2.2	LO 3.3.1	LO 4.1.1	LO 4.1.2	LO 4.2.1	LO 4.2.2	LO 4.2.3	LO 4.2.4	LO 5.1.1	LO 5.1.2	LO 5.1.3	LO 5.2.1	LO 5.3.1	LO 5.4.1	LO 5.5.1	LO 6.1.1	LO 6.2.1	LO 6.2.2	LO 6.3.1	LO 7.1.1	LO 7.1.2	LO 7.2.1	LO 7.3.1	LO 7.4.1	LO 7.5.1	LO 7.5.2
Online Discussions																																			x	x	x	x	x	x	x	x	x	x
U1: Introduction			x							x	x		x								x					x		x										x						
U2: Data Storage			x	x		x		x	x	x		x	x							x																							
U3: Program Execution						x		x	x		x	x	x	x								x								x		x	x										
U4: Operating Systems	x		x	x		x					x	x			x	x														x	x									x			
U5: Computer Networking	x	x				x	x	x				x		x			x	x																x	x	x		x	x		x		
U6: Information Security			x		x			x	x						x	x				x			x	x												x				x	x		
PT1: Explore/Impact	x	x	x	x		x														x																		x		x	x	x	x	x
U7: Algorithms and Python			x		x	x				x	x										x	x				x		x	x	x	x	x	x										
U8: Programming Languages	x	x		x	x			x		x	x	x									x	x					x		x	x	x													
U9: Software Engineering				x		x				x		x	x															x			x	x							x		x			
U10: Data Structures						x		x	x			x	x				x		x			x	x								x	x	x											
U11: Database Systems	x		x		x						x				x	x		x	x			x					x		x				x					x		x	x		x	
U12: Artificial Intelligence	x		x	x	x	x					x			x							x	x	x		x			x	x			x	x											
PT2: Create/Program		x		x	x		x			x	x										x	x					x		x	x	x	x	x											

Currently using Brookshear and Brylow (2015), *Computer Science: An Overview*

Flipped Classroom

Mon	Tue	Wed	Thu	Fri
Group Activity		Q&A		Q&A
Quiz		Finish Lab		Finish Exercises
Reading and Notes	Online Tutorial	Reading and Notes	Videos and Notes	Reading and Notes
Questions	Start Lab	Questions	Start Exercises	Questions

Assessment

- **Participation**: labs and exercises, online discussion posts, group activities
- **Quizzes**: vocab matching, multiple choice, fill in the blank, short answer
- **Explore Task**: collaborative research paper and poster presentation
- **Create Task**: pair programming assignment and individual reflection

Curriculum

Unit 1: Introduction

Algorithms, metacognition, history of computing, seven big ideas

Lab: Lightbot Hour of Code

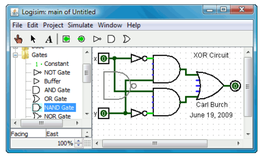
`http://lightbot.com/hocflash.html`

Unit 2: Data Storage

Logic gates, binary, hexadecimal, RAM, hard disk, ASCII, overflow

Lab: Logisim ripple carry adder

`http://www.cburch.com/logisim/`

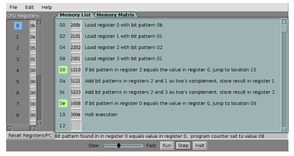


Unit 3: Program Execution

How CPUs work, instructions, machine cycle, code vs data, masking

Lab: Machine language simulator

`http://bmachine.sourceforge.net/`

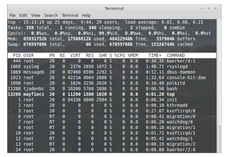


Unit 4: Operating Systems

Job scheduling, multitasking, components of OS, firmware, processes

Lab: Unix commands and files

`http://www.ee.surrey.ac.uk/Teaching/Unix/`

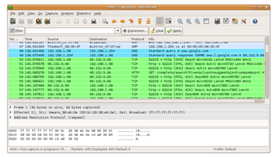


Unit 5: Computer Networking

Protocols, routers, client/server, Internet, IP, DNS, URLs, HTML

Lab: Wireshark, mtr, web tools

`https://www.wireshark.org/`

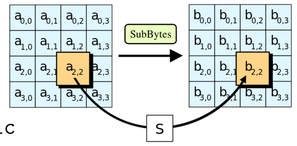


Unit 6: Information Security

Access control, privilege levels, malware, DoS, encryption, keys

Lab: Telnet vs ssh, encryption

`http://extranet.cryptomathic.com/aescalc`



Unit 7: Algorithms and Python

Primitives, pseudocode, problem solving, decisions, loop control

Lab: Intro to Python and IDLE

`http://codingbat.com/python`

Unit 8: Programming Languages

Paradigms, compiler vs interpreter, variables, functions, scope

Lab: Finch robot dance party

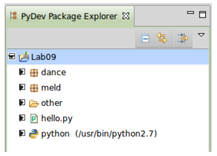
`http://www.finchrobot.com/`

Unit 9: Software Engineering

Software life cycle, prototyping, coupling, cohesion, UML diagrams

Lab: Static analysis, debugging

`http://www.aptana.com/`

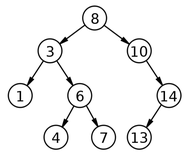


Unit 10: Data Structures

Arrays, lists, stacks, queues, trees, pointers, contiguous vs linked

Lab: Visualizing binary trees

`http://pythontutor.com/`



Unit 11: Database Systems

File system vs DBMS, schemas, relational model, SQL, data mining

Lab: Exploring your SQLite data

`http://sqlite.org/`

Unit 12: Artificial Intelligence

Turing test, semantics, production systems, state graph, heuristics

Lab: Finch robot obstacle course

`http://www.finchrobot.com/`