

Unless otherwise noted, the content of this course material is licensed under a Creative Commons Attribution 3.0 License.  
<http://creativecommons.org/licenses/by/3.0/>.

Copyright © 2009, Charles Severance.

You assume all responsibility for use and potential liability associated with any use of the material. Material contains copyrighted content, used in accordance with U.S. law. Copyright holders of content included in this material should contact [open.michigan@umich.edu](mailto:open.michigan@umich.edu) with any questions, corrections, or clarifications regarding the use of content. The Regents of the University of Michigan do not license the use of third party content posted to this site unless such a license is specifically granted in connection with particular content. Users of content are responsible for their compliance with applicable law. Mention of specific products in this material solely represents the opinion of the speaker and does not represent an endorsement by the University of Michigan. For more information about how to cite these materials visit <http://michigan.educommons.net/about/terms-of-use>.

Any medical information in this material is intended to inform and educate and is not a tool for self-diagnosis or a replacement for medical evaluation, advice, diagnosis or treatment by a healthcare professional. You should speak to your physician or make an appointment to be seen if you have questions or concerns about this information or your medical condition. Viewer discretion is advised: Material may contain medical images that may be disturbing to some viewers.

## SI502 - Midterm Review

## Details

- Will be given in discussion section - week of March 9
- Multiple Choice / Short Answer
- No more than 3 lines of technical stuff - "put code in here"
- What would this code do? What is wrong with this code?
- Notes: 2 pieces of paper - front and back is OK

## Two Topics

- Python Knowledge
- Internet Materials (2 weeks)

## Chapter I

- Input - Processing - Output
- Compiler versus Interpreter
- Programming patterns - Sequential, Conditional, Repeated and stored/reused
- Variables and Assignment Statements
- Python Comments

## Chapter 2

- Variable names / identifiers - what are legal variable names
- Reserved words - don't memorize
- Expressions / operations - Order of operation
- Input statements (input .vs. raw\_input)
- Definite loops like for i in range(10):

## Chapter 3 - Numbers

- Understand the concept of Type
- Literals have types too | 1.0 "Hello Bob"
- Operator precedence what goes first - () \* / + -
- Operators behave differently for different types
- Converting between float, integer, and string - int() float() and str()
- Math functions - No

## Chapter 4 - Strings

- Sequence of characters - indexing starts at zero
- String Literals
- String slicing x[2] x[2:4] x[3:]
- Difference between input() and raw\_input()
- The len function
- Multiplying a string - nah

## Chapter 4 - Strings

- String library - don't memorize - I will give you definitions from the book if needed
- Know about "import string"
- Know difference between split(str) and split(str, ' ')
- Opening a file
- Looping through a file for line in infile:

## Chapter 7 - Conditions

- If statements and if - else statements
- elif statements
- Indenting including nesting of if statements
- Comparison operators - know the difference between == and =
- try / except

## Chapter 8 - Loops

- Definate loops for i in range(10):
- File looping
- Loop patterns: Counting, summing, averaging, searching, detecting, maximum, minimum
- break and continue
- Understand the use of None to indicate a variable has no value at all
- Boolean Operations - and or not

## Chapter 11 - Data Structures

- Lists - append, sort, max, min - in operator
- Dictionaries - how to make empty dictionary - how to fill up a dictionary, how to use get() to initialize dictionary entries, know how to loop through dictionaries, know dictionary literals

## Inernet History

- Understanding the phases of the Internet: Research, The First Internet, Commercialization of the Internet, Ubiquity
- Understand how NSFNet was a basic change w.r.t. Society
- Review videos: Larry Smarr, Robert Cailiau, Joseph Hardin

## Internet Protocols

- Know the Internet Protocol stack and understand the purposes of each of the layers
- Know about IETF - Internet Engineering Task Force and the RFC standards
- Know about the Link/Physical layer, IP Layer, and TCP layer
- Understand how the layers make use of the services of the other layers

## Internet Protocols

- Understand IP Addresses and Domain Name addresses - how they are similar and how they are different and how they work together
- Understand network numbers
- Understand how network numbers are used in the body of the Internet for routing and how this allows the network to scale

## Internet Protocols

- Understand the nature of reliability at each of the layers - how TCP compensates for the unreliability in IP using buffering
- Understand how the traceroute program works and what it measures
- Understand how the domain name system is administered

## Internet Protocols (2)

- Understand how the Secure Socket layer works - understand what it protects from
- Understand the purpose of a TCP Port
- Know the port numbers for SMTP(Mail), Telnet, and POP (Post Office Protocol)

# HyperText Protocol (HTTP)

- Understand the Request/Response Cycle
- Be able to describe in some detail what happens when you press on a link in the browser - how the browser contacts the server, what is sent, and what comes back to the browser

# Details

- Will be given in discussion section - week of March 9
- Multiple Choice / Short Answer
- No more than 3 lines of technical stuff - “put code in here”
- What would this code do? What is wrong with this code?
- Notes: 2 pieces of paper - front and back is OK