CSCI-1010.M01: Introduction to Computer Science

Credits: 3
Semester: Fall, 2015
Times: MWF 12:00-12:50
Prerequisite: MATH-120 or 3 years of high school math
Curriculum: Core science requirement, computer science

I. Course Description and Objectives
A broad survey of the computer science discipline, focusing on the computer's role in representing, storing, manipulating, organizing, and communicating information. Topics include hardware, software, algorithms, operating systems, networks.

The main objective of this course is to provide a basic understanding of the role, activities (design, analysis, and implementation [programming] of algorithms) by the computer scientist in the modern computer. The interface of hardware, software, operating systems, and programs will be the center of study.

II. Learning Outcomes:
The student upon completion of this course will be able to:

- Explain the organization of the classical von Neumann and Turing machines and their major functional units, including logic circuits and the CPU.
- Understand binary data representation in the modern computer, including the representation of non-numeric data, and standardized file structures. Understand how fixed-length number representations affect accuracy and precision in computing.
- Identify the necessary properties of good algorithms. Discuss the importance of algorithms in the problem-solving process. Understand the software development cycle, good coding style, and algorithm development.
- Introduce the syntax of the Python programming language and develop small algorithmic programs in Python.
- Knowledge of typical algorithms used by computer scientists (for example, searching, sorting, and data compression) and O ( ) factors.
- Evaluate the choice of Abstract Data Structures – Queues, Lists, Arrays, etc. and their Python-specific implementation.

III. Textbooks and Readings

Recommended: For a centralized location of introductory topic material, the textbook available in the library:
Computer Science Illuminated, by Nell Dale and John Lewis, 5th Edition.

Other:
Selected Internet readings as given on http://bob-europe.com/moodle, and on the syllabus.

IV. Instructor Information

Name	Kirk Tennant
Web site	http://bob-europe.com/moodle
E-mail:	tennantk@slu.edu
Office Location	#34 2nd Floor, Business and Economic Faculty Salon
Office Hours	MWF 10:00-12:00
Phone	91 554 58 58 (254)
Biography	Degrees: MS (Management) MA Theological Studies MS (Engineering Management) MA (English Arts)

V. Course Outline (5 sections)

i. Introduction to Computing

Week 1: Welcome to the Digital Age, Number Systems: Base Arithmetic, Computer Mathematics, Artificial Intelligence
Week 2: Data**,** Information, and It's Computer Representation/**Project #1**
Week 3: Working with Computer Information and Physical Implementation of Data and Information in Computers
Week 4: Computer Hardware and Architecture
Week 5: Operating Systems; What is an Operating System? Parts of an Operating Systems; Basic Linux and Windows Commands/**Project #2**
Week 6: Introduction to Networking

ii. Introduction to Computer Programming

Week 7: Universal Abstract Data Types/ **Midterm Exam**
Week 8: Introduction to Python Programming Language
Week 9: Problem Solving and Algorithm Implementation in Python

iii. Understanding, Encryption Algorithms and File and Network Security

Week 10: Data Representation - Binary Trees /Huffman Coding/
Week 11: Advanced Encryption Algorithms/**Project #3**

iv. Introduction to Simulation and Modelling Languages

Week 12: Using Excel in Engineering Simulations
Week 13: Advanced Engineering Simulations/**Project #4**

v. Android Applications and Programming

Week 14: Using GPS Location Devices and Implementing an Android App in AppInventor2/**Project #5**
Week 15: Using GPS Location Devices and Implementing an Android App in AppInventor2/**Final Exam**

VI. Grading System

The grade will be obtained from the following areas:

Homework (6):	30%
Projects (5):	30%
First Mid-Term Exam:	20%
Final Exam:	20%

Both Homework and Project assignments are made to be completed by the next week.

Any late work will be penalized at 5% per day late. If for example homework was due on Tuesday, but not submitted until Thursday the maximum grade for a 100% correct paper will be 90%.

Grading scale
100 < A < 95%,
95% <A- < 90%
90% < B+ < 87%
87% < B < 83%
83% < B-< 80%
80% < C+ < 77%
77% < C < 73%
73% < C- < 70%
70% < D < 50%
F < 50%

VII. Important Dates

The last date to withdrawal from a course without a grade or W is <u>September 14, 2015</u>. Last day to drop with a W is October 29, 2015. If you miss this deadline and are registered for the class at the end of the semester without completing the required work, you <u>will</u> be assigned a letter grade of F. The final examination for this class is <u>Tuesday, December 15, 2015 at 12:00.</u>

VIII. Academic Honesty and Plagiarism

The University is a community of learning, whose effectiveness requires an environment of mutual trust and integrity, such as would be expected at a Jesuit, Catholic institution. As members of this community, students, faculty, and staff members share the responsibility to maintain this environment. Academic dishonesty violates it. Although not all forms of academic dishonesty can be listed here, it can be said in general that soliciting, receiving, or providing any unauthorized assistance in the completion of any work submitted toward academic credit is dishonest. It not only violates the mutual trust necessary between faculty and students but also undermines the validity of the University's evaluation of students and takes unfair advantage of fellow students. Further, it is the responsibility of any student who observes such dishonest conduct to call it to the attention of a faculty member or administrator.

Examples of academic dishonesty would be copying from another student, copying from a book or class notes during a closed-book exam, submitting materials authored by or editorially revised by another person but presented as the student's own work, copying a passage or text directly from a published source without appropriately citing or recognizing that source, taking a test or doing an assignment or other academic work for another student, tampering with another student's work, securing or supplying in advance a copy of an examination without the knowledge or consent of the instructor, and colluding with another student or students to engage in an act of academic dishonesty.

Where there is clear indication of such dishonesty, a faculty member or administrator has the responsibility to apply appropriate sanctions. Investigations of violations will be conducted in accord with standards and procedures of the school or college through which the course or research is offered. Recommendations of sanctions to be imposed will be made to the dean of the school or college in which the student is enrolled. Possible sanctions for a violation of academic integrity include, but are not limited to, disciplinary probation, suspension, and dismissal from the University. The complete SLU Academic Honesty Policy can be found at the following link: http://spain.slu.edu/academics/academic_advising/docs/Academic_integrity.pdf

IX. Specific Course Policies

(1) Students are encouraged to participate in class discussions and to ask questions.
(2) Announcements may be made during the semester which alter the course content.
(3) Useful information, projects, and homework for the course may be found on the course Moodle site. http://bob-europe.com/moodle. Students should register for this course at the Moodle site at once.
(4) Syllabus, reading and homework problems are subject to change.
(5) Students are responsible for all lecture material, handouts, homework and assigned reading.
(6) It is mandatory to attend all classes unless a reasonable excuse is given.
(7) Make up exams are not given. Students who legitimately miss an exam, due to a doctor's visit or family emergency must provide written documentation of the circumstances. A letter from the university counselor is accepted. Exams that are missed illegitimately result in a score of F. Grades for these students will be based on the remaining exams. Missing more than one exam always results in an F grade.

X. Accommodation Statement

In recognition that people learn in a variety of ways and that learning is influenced by multiple factors (e.g., prior experience, study skills, learning disability), resources to support student success are available on campus. Students who think they might benefit from these resources can find out more about:

1. Course-level support (e.g., faculty member, departmental resources, etc.) by asking your course instructor.
2. University-level support (e.g., tutoring/writing services, Disability Services) by visiting the Academic Dean's Office (San Ignacio Hall) or by going to http://spain.slu.edu/academics/learning_resources.html.
3. Students who believe that, due to a disability, they could benefit from academic accommodations are encouraged to contact Disability Services at +34 915 54 58 58, ext. 204, send an e-mail to counselingcenter-madrid@slu.edu, or to visit the Counseling Office (San Ignacio Hall). Confidentiality will be observed in all inquiries. Course instructors support student accommodation requests when an approved letter from Disability Services has been received and when students discuss these accommodations with the instructor after receipt of the approved letter.

.

Detailed Course Outline
CSCI 140 Assignments
TTH 17:30 – 18:50

Day	Date	Topic	Web Sites/Readings
1	1/9	**Introduction to Number Systems/Artificial Intelligence** • Introduction to Digital Age/Information • Positional notation • Finite number sequences – single binary digit (bit), four binary digits (nibble), eight binary digits (Byte) • Use of different number systems – specifically base 2, Base 8, Base 16 with reference to Base 10 • Addition, Subtraction, Multiplication, Division in ALL Bases • Fractions with Bases • Finite representation errors generated by Base Conversion	http:// http://www.mazeworks.com/hex7/ http://www.howstuffworks.com/mp3.htm http://courses.cs.vt.edu/~csonline/NumberSystems/Lessons/index.html
2	3/9	**Introduction to Number Systems** • Integer Representation • Floating-point Representation (Decimal point = radix point) • Multiplication, Division in ALL Bases • Problems with negative numbers • One's Complement Math • Two's Complement Math	**HOMEWORK #1 ASSIGNED** http://courses.cs.vt.edu/~csonline/NumberSystems/Lessons/index.html
3	8/9	**Understanding What Constitutes Information** • Introduction to Information • Representing Text • ASCII Character Set • Unicode Character Set • Text Compression • Keyword encoding • Run-length encoding (extension to graphic files - JPG)	Chapter7of *The Information: History, Theory, Flood* http://astarte.csustan.edu/~tom/SFI-CSSS/info-theory/info-lec.pdf http://www.fieggen.com/ian/g_formats.htm

4	10/9	**Understanding What Constitutes Information** • Huffman Coding Schemes (variable length binary encoding) • Audio Formats – WAV, AU, AIFF, VQF, MP3, MP4 • Images and Graphics - JPG,GIF, TIFF, PNG • Representing Color RGB, CYMK • Vector Images - Illustrator, Flash • Video Formats – CODECS (Compressor/Decompression) Real Video or MPEG2, MPEG4	**PROJECT #1 ASSIGNED** http://sinus.if.pw.edu.pl/podziemski/wp-content/uploads/downloads/2012/05/jpegtool.pdf http://www-msp.ece.mcgill.ca/documents/audioformats/index.html https://ccrma.stanford.edu/courses/422/project/WaveFormat/
5	15/9	**Translating Information to the Physical Computer** • Introduction to Gates (Only 2 inputs) • Half-Adder Circuit • Full-Adder Circuit • Memory Circuits • CPU and GPU Chips	http://www.ee.surrey.ac.uk/Projects/Labview/gatesfunc/index.html http://inst.eecs.berkeley.edu/~ee42/sp04/lectures/lecture14student.pdf
6	17/9	**Translating Information to the Physical Computer** • Using Circuit Design Software	**FIRST MIDTERM ASSIGNED** http://www.ee.surrey.ac.uk/Projects/Labview/gatesfunc/index.html http://home.anadolu.edu.tr/~atdogan/EEM232/06-CombCktDesign.ppt
7	22/9	**Understanding Computer Architecture and Components** • CPU • Memory • BIOS • Motherboard • Ports and Slots	http://www.csupomona.edu/~hnriley/www/VonN.html http://mathworld.wolfram.com/TuringMachine.html http://www.it.iitb.ac.in/~shourya/docs/hardware.pdf

8	24/9	**Understanding Computer Architecture and Components** • How the CPU Works • Machine Language • Assembler Language • High-Level Languages – Interpreted or Compiled • Super Simple CPU Software • Implementing Simple Algorithms (GCM, LCM)	**PROJECT #2 ASSIGNED** http://www.youtube.com/watch?v=cNN_tTXABUA
9	29/9	**Introduction to Operating Systems**	http://courses.cs.vt.edu/~csonline/OS/Lessons/Introduction/index.html http://www.cmpe.boun.edu.tr/~uskudarli/courses/cmpe235/os.pdf
10	1/10	**Introduction to Operating Systems**	http://courses.cs.vt.edu/~csonline/OS/Lessons/Introduction/index.html http://www.cmpe.boun.edu.tr/~uskudarli/courses/cmpe235/os.pdf
11	6/10	**Introduction to Networking** Hardware	
12	8/10	**Introduction to Networking** Software and Protocols	
13	13/10	**Understanding Abstract Data Types** • Queues • Arrays • Lists • Linked Lists • Trees	**HOMEWORK #3 ASSIGNED** http://www.idevelopment.info/data/Programming/data_structures/overview/Data_Structures_Algorithms_Introduction.shtml http://cslibrary.stanford.edu/103/LinkedListBasics.pdf http://python.dzone.com/articles/algorithm-week-binary-search-0
14	15/10	**Understanding Abstract Data Types Functions** • Sorting • Filtering	http://ww3.algorithmdesign.net/handouts/MergeSort.pdf http://www.topcoder.com/tc?module=Static&d1=tutorials&d2=sorting
15	20/10	**Programming in Python** • Introduction to the General Structure of Python • General Data Types in Python • Using the Python Script and Interactive Modes	**Thinking as a Computer Scientist in Python** http://heather.cs.ucdavis.edu/~matloff/Python/PythonIntro.pdf

16	22/10	**Programming in Python** • Implementing Abstract Data Types in Python – Dictionaries, Lists, Tuples • Control Statements in Python	**HOMEWORK #4 ASSIGNED** **Thinking as a Computer Scientist in Python** http://heather.cs.ucdavis.edu/~matloff/Python/PythonIntro.pdf
17	27/10	**Problem Solving, Algorithms, and Programming in Python**	**Thinking as a Computer Scientist in Python** **PROJECT #3 ASSIGNED** http://www.huffmancoding.com/david/algorithm.html
18	29/10	**Problem Solving, Algorithms, and Programming in Python**	**Thinking as a Computer Scientist in Python** http://www.huffmancoding.com/david/algorithm.html
19	3/11	**Binary and Other Trees** In-class Project: Implementing Huffman Coding in Python	
20	5/11	**Binary and Other Trees** In-class Project: Implementing Huffman Coding in Python	
21	10/11	**Understanding Today's Encryption Algorithms and Protecting File Data** • Asymmetric Keys (Public/Private) • RSA Algorithm	**PROJECT #3 ASSIGNED** http://www.esat.kuleuven.be/cosic/intro/
22	12/11	**Understanding Today's Encryption Algorithms and Protecting File Data** • Asymmetric Keys (Public/Private) • RSA Algorithm	http://www.esat.kuleuven.be/cosic/intro/
23	17/11	**Introduction to Simulation and Modeling Languages for Computer Scientists and Engineers** • Model Structures in Excel • Introduction to Excel • AutoFill • Formulas • Functions	http://oit.wvu.edu/training/files/excel2010_intro.pdf http://bookboon.com/en/excel-2010-introduction-part-i-ebook

24	19/11	**Introduction to Simulation and Modeling Languages for Computer Scientists and Engineers** Waiting Line (Queue) Models in Excel	**HOMEWORK #5 ASSIGNED** Simpler Spreadsheet Simulation of Multi-Server Queues
25	24/11	**Introduction to Simulation and Modeling Languages for Computer Scientists and Engineers**	**http://heather.cs.ucdavis.edu/~matloff/156/PLN/DESimIntro.pdf**
26	26/11	**Introduction to Simulation and Modeling Languages for Computer Scientists and Engineers**	**http://heather.cs.ucdavis.edu/~matloff/156/PLN/DESimIntro.pdf**
27	1/12	**Using GPS Location Devices and Android** • Introduction to AppInventor • Introduction to GPS technology	Step by Step AppInventor code.google.com/p/android-scripting code.google.com/p/android-scripting/wiki/SharingScripts
28	3/12	**Using GPS Location Devices and Android** • Introduction to AppInventor • Introduction to GPS technology	Step by Step AppInventor code.google.com/p/android-scripting code.google.com/p/android-scripting/wiki/SharingScripts
29	8/12	**Public Holiday – No Class**	
30	10/12	**Using GPS Location Devices and Android** • Introduction to AppInventor • Introduction to GPS technology	**PROJECT #5 ASSIGNED** AppInventor Book code.google.com/p/android-scripting/wiki/ApiReference
		Final Examination Due	**Tuesday, December 15, 2015 at 12:00.**