COURSE SYLLABUS

CPSC1301 – COMPUTER SCIENCE I - FALL 2016

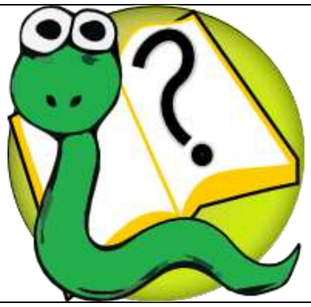
INSTRUCTOR INFORMATION

INSTRUCTOR NAME: Dr. Vladimir Zanev
EMAIL: zanev_vladimir@columbusstate.edu
PHONE: (706) 507-8182
OFFICE HOURS AND LOCATION: MWF 2:00 p.m. – 4:00 a.m., TR 10:30 a.m. – 12:00 p.m., CCT 442
MEETING TIME AND PLACE: TR, 1:30 p.m. – 2:45 p.m., CCT 406
Appointments for a meeting outside of these hours can be made at: zanev_vladimir@columbusstate.edu
COURSE CRN NUMBER/TITLE: CPSC 1301 Computer Science I (CRN 82000)
CREDIT HOURS/PREREQUISITES: 3 credit hours, co-requisite: CPSC 1301L

COURSE DESCRIPTION: This course includes an overview of computers and programming; problem solving and algorithm development; simple data types; arithmetic and logic operators; selection structures; repetition structures; text files; arrays (one-and-two-dimensional); procedural abstraction and software design; modular programming (including sub-programs or the equivalent).

REQUIRED TEXTBOOK AND MATERIALS

1.	http://www.zyBooks.com	**An interactive online book** Title: Programming in Python 3.3 Authors: B. Miller, F. Vahid, R. Lysecky Publisher: zyBooks Edition: March 2015 http://www.zybooks.com
	To subscribe for the online book: 1. Sign up at zyBooks.com 2. Enter zyBook code: COLUMBUSSTATECPSC1301Fall2016 3. Click Subscribe (you will be expected to pay next) 4. The cost to subscribe is $48 The student subscription will be valid through 12/20/2016. View the zybook registration video at: http://vimeo.com/135692064	

2.		**Online book (free)** Title: How to Think Like a Computer Scientists. Learning with Python 3 Authors: P. Wentworth, J. Elkner, A. Downey, C. Meyers Version Date: November 2011 Available at URL (html): http://openbookproject.net/thinkcs/python/english3e/ Download pdf at: http://cs.nyu.edu/courses/spring13/CSCI-UA.0002-001/HowToThinkLikeAComputerScientist_%20LearningWithPython3.pdf

RECOMMENDED TEXTBOOKS / WEB RESOURCES

WEB RESOURCES

The Python Tutorial	https://docs.python.org/3/tutorial/index.html
Python Beginner's Guide	https://wiki.python.org/moin/BeginnersGuide
Python 3 Tutorial	http://www.python-course.eu
Official Python Documentation	https://docs.python.org/3/

E-BOOKS

Dive Into Python	Recommended to be one of the beginning books to build your Python basics.
A Byte of Python	Another book for a beginner who wants an easy and progressing way of learning Python.

SUPPLEMENTARY BOOKS AND MATERIALS

- 1301 Blog: https://csdcsblogs.wordpress.com/
- Cougarview materials (https://colstate.view.usg.edu/)
- Class handouts & notes
- Software and manuals found in the computer labs and on the Internet

LEARNING OUTCOMES

Course Objectives: Upon completion of this course, students will

- demonstrate an understanding of fundamental concepts of computer programming, including problem solving and algorithm development, procedural abstraction and software design
- be able to use simple data types, control structures, array and string data structures
- demonstrate these basic skills in programming by writing several significant programs
- use good software engineering principles and developing fundamental programming skills in the context of a language that supports the object-oriented paradigm
- be familiar with standard debugging techniques.

Course Outcomes:

- The students will demonstrate an understanding of computing and computer science.
    - Strategies and Actions used to produce the outcome:
        - Study concepts of computing and computer science.

        - Class discussion about what is computing and computer science.
    - ABET Criteria covered: A, E, and G
    - Program Objectives covered: 2
    - Assessment Methods: Written Assignments, Quizzes, and Exams.
- The students will demonstrate knowledge of computer programming concepts.
    - Strategies and Actions used to produce the outcome:
        - Study concepts of computer programming.
        - Read and write moderately complex programs in a programming language.
    - ABET Criteria covered: A, B, and C
    - Program Objectives covered: 2 and 3
    - Assessment Methods: Written and Programming Assignments, Quizzes, and Exams.
- The students will demonstrate knowledge of basic syntax of a specific programming language
    - Strategies and Actions used to produce the outcome:
        - Study of concepts of computer programming.
        - Study the API for the programming language.
    - ABET Criteria covered: A, B, and C
    - Program Objectives covered: 2 and 3
    - Assessment Methods: Written and Programming Assignments, Quizzes, and Exams.
- The students will demonstrate the ability to read moderately complex programs written in a specific programming language and understand what these programs do
    - Strategies and Actions used to produce the outcome:
        - Study concepts of computer programming.
        - Read and write moderately complex programs in a programming language.
    - ABET Criteria covered: A, B, and C
    - Program Objectives covered: 2 and 3
    - Assessment Methods: Written and Programming Assignments, Quizzes, and Exams.
- The students will demonstrate the ability to design algorithms utilizing the principles of object-oriented programming to solve moderately complex problems
    - Strategies and Actions used to produce the outcome:
        - Study concepts of computer programming.
        - Read and write moderately complex programs in a programming language.
    - ABET Criteria covered: A, B, and C
    - Program Objectives covered: 3
    - Assessment Methods: Written and Programming Assignments, Quizzes, and Exams.
- The students will demonstrate the ability to write moderately complex programs in a specific programming language to implement these algorithms
    - Strategies and Actions used to produce the outcome:
        - Study concepts of computer programming.
    - ABET Criteria covered: A, B, and C
    - Program Objectives covered: 3
    - Assessment Methods: Written and Programming Assignments, Quizzes, and Exams.
- The students will demonstrate the ability to follow specified style guidelines in writing programs, and understand how the guidelines enhance readability and promote correctness in programs
    - Strategies and Actions used to produce the outcome:
        - Study concepts of computer programming.
    - ABET Criteria covered: A, B, C, and J
    - Program Objectives covered: 2 and 3
    - Assessment Methods: Written and Programming Assignments, Quizzes, and Exams.
- The students will demonstrate the ability to edit, compile, debug and run programs in a specific programming language
    - Strategies and Actions used to produce the outcome:
        - Study concepts of computer programming.

- ABET Criteria covered: A, B, and C
- Program Objectives covered: 2 and 3
- Assessment Methods: Written and Programming Assignments, Quizzes, and Exams.

COURSE ASSESSMENT

LEARNING ACTIVITIES

Class Sessions
The CPSC 1301 class will meet TR from 1:30 p.m. to 2:45 p.m. at CCT 406 for two seventy five minute sessions each week. The class sessions will focus on Python programming language, Python program development, Python programming, and problem solving. A class session is a lecture supposed to make an introduction to topics using the textbooks (Zy text and How To … text), lecture notes, and Python source code. Make your own notes. They help you to learn and memorize basic concepts and notions and you more successful with the quizzes, programming assignments and exams. The lecture topics covered in the class follow the course schedule. Each student is expected to attend all class lectures, read the textbook chapters, make notes, and participate in the class discussions.

Quizzes
The quizzes are designed to help students learn better the course Python programming and problem-solving topics and prepare well for the exams. Eight quizzes are scheduled covering the main topics of the course. The quizzes are one attempt, closed textbooks, in-class quizzes. Quizzes include true/false, multiple-choice, fill-in, and programming problem questions that are related to the corresponding textbook chapter topics.

Assignments
Programming activities as problem-solving using Python programming language, designing and developing Python source code, compiling and debugging, running and testing of Python programs cannot be learned simply by reading a textbook. The assignments are "hands-on practice" part of the course that allows developing skills and experience in implementing algorithms and data structures in Python programs. You must practice, practice, and practice solving different problems by implementing real programs. Assignments will focus on one or more of the learning objectives. Eleven programming problem-solving assignments developing Python programs will be assigned and graded. The assignments have to be completed no later than 11:59 p.m. on the due day (see the class schedule).**Late assignments are not accepted for credits**.

Challenge Activities
The Challenge Activities are homework programming exercises from the ZyBook. They are Web-based, Python programming solving problems. The Challenge Activity programming environment gives you immediate access to the Python interpreter and the correct Python program solution is registered on the zyBook site. Eight Challenge Activities have to be completed. The Challenge Activities have to be completed no later than 11:59 p.m. on the due day (see the class schedule).**Late Challenge Activities are not accepted for credits**.

Midterm and Final Exam
Your performance in this class will be measured by two exams - Midterm and Final Exam**.** The Midterm and the Final Exam will be problem-solving, timed, take-home, programming exams. During the exams you have to solve problems writing Python programs. **No make-up exam will be given unless the exam was missed due to a documented emergency.**

The ACM recommends the following: “As a general guideline, the amount of out-of-class work is approximately ***three times*** the in-class time. Thus, a unit that is listed as requiring 3 hours typically entails a total of 12 hours (3 in class and 9 outside).” Students will be expected to spend this time outside class reading the books, lecture notes and other materials; writing solutions to homework exercises and programming assignments.
From A Study Guide: “Study actively. Ask yourself questions, review your notes regularly, create concept maps, and discuss key concepts with peers and your instructor. FACT: Association is a key to memory and cognitive research has

shown that you will remember 10 percent of what you read, 20 percent of what you hear, 30 percent of what you see, 50 percent of what you hear and see together, 70 percent of what you say, and 90 percent of what you do!"
A Word about Learning to Write Computer Programs: You CANNOT learn to program by just listening to how it is done and reading through examples in the lectures and text book. YOU MUST WRITE CODE!

COURSE EVALUATION

The final grade will be obtained from the following:

Quizzes	15%
Challenge Activities	15%
Programming Assignments	30%
Midterm Exam	20%
Final Exam	20%

Class Participation/Attendance will be considered for students who are on the borderline between two letter grades (0-2%).

The letter grade will be assigned as follows:

Approx. Percentage Range	**Final Grade**	
90-100%	A	• fulfills or exceeds all of the assigned content requirements. • knowledge of the subject is accurate throughout • exhibits convincing range and quality of knowledge, having done appropriate research, if applicable.
80-89%	B	• fulfills all of the important assigned content requirements • knowledge of the subject is accurate throughout except in minor details. • seems informed on the subject, having done appropriate research, if applicable
70-79%	C	• fulfills most of the important assigned content requirements. • knowledge of the subject is generally accurate, though flawed • exhibits limited range or quality of knowledge, having done limited appropriate research, if applicable.
60-69%	D	• fulfills some of the important assigned content requirements • knowledge of the subject is generally accurate, though flawed • exhibits limited range or quality of knowledge, having done

		minimal appropriate research, if applicable.
59% and below	F	• fails to address the important requirements of the course. • knowledge of the subject is generally inaccurate and/or lacks range or quality

ADMINISTRATIVE POLICIES AND ACADEMIC RESOURCES

CSU DISABILITY POLICY

If you have a documented disability as described by the Americans with Disabilities Act (ADA) and the Rehabilitation Act of 1973, Section 504, you may be eligible to receive accommodations to assist in programmatic and/or physical accessibility. We recommend that you contact the Office of Disability Services located in Schuster Student Success Center, Room 221, 706-507-8755 as soon as possible. Students taking online courses can contact the Office of Disability services at http://disability.columbusstate.edu/. The Office of Disability Services can assist you in formulating a reasonable accommodation plan and in providing support. Course requirements will not be waived but accommodations may be able to assist you to meet the requirements. Technical support may also be available to meet your specific need.

ACADEMIC INTEGRITY

All students are expected to recognize and uphold standards of intellectual and academic integrity. As a basic and minimum standard of conduct in academic matters that students be honest and that they submit for credit only the products of their own efforts. Both the ideals of scholarship and the need for fairness require that all dishonest work be rejected as a basis for academic credit. They also require that students refrain from any and all forms of dishonorable or unethical conduct related to their academic work.

Students are expected to comply with the provisions of Section III, "Student Responsibilities," of the Columbus State University Student Handbook. This specifically includes the sections on "Academic Irregularity," and "Conduct Irregularity." In particular, the Columbus State University Student Handbook states:

“No student shall give or receive assistance in the preparation of any assignment, essay, laboratory report, or examination to be submitted as a requirement for any academic course in such a way that the submitted work can no longer be considered the personal effort of the student submitting the work.”

Examples of Academic Dishonesty include but are not limited to: Plagiarism (see definition below), giving or receiving unauthorized assistance on exams, quizzes, class assignments or projects, unauthorized collaboration, multiple submissions (in whole or part) of work that has been previously submitted for credit.

Plagiarism is any attempt to represent the work or ideas of someone else as your own. This includes purchasing or obtaining papers from any person and turning them in as your own. It also includes the use of paraphrases or quotes from a published source without properly citing the source. All written assignments may be submitted for textual similarity review to Turnitin.com for the detection of plagiarism.

Any work turned in for individual credit must be entirely the work of the student submitting the work. **All work must be your own.** You may share ideas but submitting identical assignments (for example) will be considered cheating. **You may discuss the material in the course and help one another with debugging, however, I expect any work you hand in for a grade to be your own.** A simple way to avoid inadvertent plagiarism is to talk about the assignments, but don't read each other's work or write solutions together. Keep scratch paper and old versions of assignments until after the assignment has been graded and returned to you. **If you have any questions about this, please see me immediately.**

For assignments, access to notes, textbook, books and other publications is allowed. Stealing, **giving or receiving** any code, diagrams, drawings, text or designs from another person (CSU or non-CSU) is not allowed. Having access to another person's work on the system or giving access to your work to another person is not allowed. It is your responsibility to keep your work confidential.

No cheating in any form will be tolerated. Please be aware that anyone caught cheating or plagiarizing in this class will receive a "0" for the assignment/exam and may receive a "0" for the course.

STUDENT COMPLAINT PROCESS

Information and resources for student complaints and academic appeals are located at the following link on the Columbus State University website http://aa.columbusstate.edu/appeals/.

COURSE ATTENDANCE POLICY

Class attendance is the responsibility of the student, and it is the student's responsibility to independently cover any materials missed. Class attendance and participation may also be used in determining grades. It is your responsibility to sign a roll sheet for every class meeting. At my discretion, I may drop you from the course for more than ***six (6)*** absences. Missing an exam or quiz is considered an absence. Missing labs cannot be made up unless you have documented proof of an emergency that includes contact information for verification. Missed classes caused by participation in documented, formal, University-sponsored events or mandatory military events will not count as absences provided you notify me of such anticipated absences in advance and as soon as possible.

You are responsible for all class work missed, regardless of the reason for the absence(s). Late assignments will **not** be accepted, so if you are absent, it is your responsibility to make alternate arrangements. No makeup exams or quizzes will be given, so please make sure you are present for all exams/quizzes. Refer to the CSU Catalog (http://ace.columbusstate.edu/advising/a.php#AttendancePolicy) for more information on class attendance and withdrawal.

Electronic Devices and Academic Integrity: All cell phones and pagers must be turned off prior to entering the classroom or lab. The use of any electronic device during a test or quiz is prohibited. This includes cell phones, handheld calculators, iPhones, Android phones, PalmPilots, Blackberrys, PocketPCs, and laptops. Any use of such a device during a test or quiz will be considered a breach of academic integrity.

TECHNICAL RESOURCES

HARDWARE REQUIREMENTS
How do I know if my computer will work with D2L?

SOFTWARE REQUIREMENTS
An- office suite such as Microsoft Office or Open Office

- To open PDF files you might need Acrobat Reader
- Browser Plugins (Pdf files, QuickTime files, Mp4 files) can be usually be obtained at the browsers website.
  Google Chrome
  Firefox
  Safari
  Internet Explorer (Caution: IE is often problematic for D2L-CougarVIEW)
- Python IDE (IDLE – recommended, PyCharm, Wing IDE, jGrasp)

If you need technical support or need assistance configuring your computer, you can refer to the link located in the "Support Resources" widget located on your "My Home" and your "Course Home" pages. If you cannot solve your problem after reviewing the knowledge base help pages, you can call help center 24-7 and talk to a Help Center agent. The number is 1-855-772-0423.

Course Material Downloads

Python (with IDLE): http://www.python.org/download/
PyCharm Community IDE: http://www.jetbrains.com/pycharm/
Wing 101 IDE Personal: http://wingware.com/downloads/wingide-personal/
JGrasp: http://www.jgrasp.org/

OTHER

How to Access the Course

You can access the course through CougarView at: http://colstate.view.usg.edu/

At this page, select the "Log on to" CougarView link to activate the CougarView logon dialog box, which will ask for your CougarView username and password. Your CougarView username and password are the same as your Cougarnet username and password:

Username: lastname_firstname
Password: XXXX
Default password is your birthday in the format of DDMMYY.

If you try the above and CougarView will not let you in, please use the "Comments/Problems" link on the CougarView home page to request help. If you are still having problems gaining access a day or so after the class begins, please e-mail me immediately.

Once you've entered CougarView, you will see a list of courses you have access to. The CPSC 1301L course is listed as "Computer Science 1 Lab". Next to this, you should see my name as the instructor. You may also see new discussion postings, new calendar postings, and new mail messages. Clicking on the name of the course will take you to the course's home page. If you do not see the "**Computer Science 1 Lab**" course in the list, please e-mail me **immediately**.

Once you have clicked on the course's name and accessed the particular course itself, you will find a home page with links to other sections and tools, and a menu on the left-hand side. Feel free to explore the areas in the course.

Course Website

It is your responsibility to frequently look at the CougarView course website to keep your knowledge of class activities current. I may occasionally forget to announce details in class, but they may have been already posted on the site and/or in CougarView. If so, you will still be held responsible for them. For example, assignment due dates, corrections of errors, announcements, exam dates, changes to policies, and so on.

Getting help

Student assistants in the public Computer Center labs / Library can help you with basic computer-related problems such as logging on to the network, saving your work, etc., but they are not obligated to help you with your assignments. There are several tutors in the School of Computer Science lab (CCT450) who can help you with the assignments. Their schedule is posted in the Computer Science School. There is also an online tutoring in D2L run by graduate computer science students. The tutoring will appear in your course list in cougar view. You can post questions in the discussion forum and they will be answered in a timely fashion. You can also email a particular tutor with problems with code.

Discussion Etiquette

CSU is committed to open, frank, and insightful dialogue in all of its courses. Diversity has many manifestations, including diversity of thought, opinion, and values. Students are encouraged to be respectful of that diversity and to refrain from inappropriate commentary. Should such inappropriate comments occur, I will intervene as I monitor the dialogue in the discussions. I will request that inappropriate content be removed from the discussion and will recommend university disciplinary action if deemed appropriate. Students as well as faculty should be guided by common sense and basic etiquette. The following are good guidelines to follow:

- Never post, transmit, promote, or distribute content that is known to be illegal.
- Never post harassing, threatening, or embarrassing comments.
- If you disagree with someone, respond to the subject, not the person.

Never post content that is harmful, abusive; racially, ethnically, or religiously offensive; vulgar; sexually explicit; or otherwise potentially offensive.

Student Responsibilities

As a student in this course, you are responsible to:

- manage your time and maintain the discipline required to meet the course requirements,
- come to class prepared to ask questions to maximize your understanding of the material,
- complete all readings,
- complete all assignments,
- complete all quizzes and exams,
- actively participate in discussions,
- submit the “one-minute paper” after each class, and
- read any e-mail sent by the instructor and respond accordingly.

“I didn’t know” is **NOT** an acceptable excuse for failing to meet the course requirements. If you fail to meet your responsibilities, you do so at your own risk.

Instructor Responsibilities

As your instructor in this course, I am responsible to:

- lead the class discussion and answer students’ questions,
- post weekly lessons outlining the assignments for the week,
- read all responses to discussion questions and comments to responses,
- actively participate in discussions when necessary,
- respond to students questions and concerns expressed in the “one-minute paper”,
- grade assignments, quizzes, and exams, and post scores within one week of the end of the week in which they are submitted, and
- read any e-mail sent by the you and respond accordingly within 48 hours.

Although I will read every posted discussion question and response, I will not necessarily respond to every post.

Student Portfolio

Students are encouraged to keep and maintain a portfolio of all of their work (notes, source code, assignments, etc.) throughout their academic program. It is recommended that you keep a copy on your personal H: drive at CSU and back it up regularly on your own portable media or in the cloud.

TENTATIVE COURSE SCHEDULE

The schedule is based on 27 days of 85-minute online MTWR classes and 7 weeks of instruction. You have to cover all scheduled class work and requirements on strict basis session by session, week by week (readings, lecture notes, assignments, quizzes, challenge activities, and exams). **All due dates and times for the class requirements are firm.**

Lecture Topics and References

- Zy – ZyBook Programming in Python 3.3
- How – : How to Think Like a Computer Scientists. Learning with Python 3
- LN – Lecture Notes (available on the class CougarView site)

WEEK	DATE	TOPICS	QUIZZES, ASSIGNMENTS, EXAMS
1	**August 15th – 19th**		
	Tue, 08/16	Introduction to the course. Class organization and administration. Problem Solving/Computational Thinking (Ch. 1.1 , 1.2 Zy, LN)	
	Thu, 08/18	Problem Solving/Computational Thinking (Ch. 1.1 , 1.2 Zy, LN)	
2	**August 22nd – 26th**		
	Tue, 08/23	Problem Solving/Computational Thinking (Zy, LN) Introduction to Computers, Programs, and Python (Ch 1 Zy, Ch 1 How, LN) Introduction to Computers, Programs, and Python (Ch 1 Zy, Ch 1 How, LN)	
	Thu, 08/25	Introduction to Computers, Programs, and Python (Ch 1 Zy, Ch 1 How, LN)	Quiz 1. Computational Thinking. Computers and Programs Assignment 1 Challenge Activities. Chapter 1 Zy
3	**August 29th – September 2nd**		
	Tue, 08/30	Elementary Programming: Variables, Expressions, Data Types. Strings. Input, Output. (Ch 2,3 Zy, Ch 2 How)	Quiz 2. Introduction to Python and IDLE
	Thu, 09/01	Elementary Programming: Variables, Expressions, Data Types. Strings. Input, Output. (Ch 2,3 Zy, Ch 2 How)	
4	**September 5th – 9th**		
	Tue, 09/06	Simple Turtle Graphics (Ch 3 How)	Quiz 3. Elementary Programming Assignment 2. Elementary Programming Challenge Activities. Chapter 2 and 3 Zy
	Thu, 09/08	Simple Turtle Graphics (Ch 3 How)	
5	**September 12th – 16th**		

	Tue, 09/13	Relational Operators Decisions: if-else Boolean Variables and Operators Nested Branches (Ch 4 Zy, Ch 5 How)	Assignment 3. Turtle Graphics
	Thu, 09/15	Relational Operators Decisions: if-else Boolean Variables and Operators Nested Branches (Ch 4 Zy, Ch 5 How)	
6	**September 19th – 23rd**		

	Tue, 09/20	*for* Loop
		while Loop
		Problem Solving: Hand-tracing
		(Ch 5 Zy, Ch 7 How)
			Quiz 4. Branches
			Assignment 4. Branches
			Challenge Activities. Chapter 4 Zy

	Thu, 09/22	*for* Loop
		while Loop
		Problem Solving: Hand-tracing
		(Ch 5 Zy, Ch 7 How)	

7	**September 26th – 30th**		
	Tue, 09/27	Nested Loops Tables Input Validation (Ch 5 Zy, Ch 7 How)	Quiz 5. Loops Assignment 5. *for* and *while* Loops Challenge Activities. Chapter 4 Zy
	Thu, 09/29	Nested Loops Tables Input Validation (Ch 5 Zy, Ch 7 How)	
8	**October 3rd – 7th**		
	Tue, 10/04	Review for the Midterm Exam	Assignment 6. Nested Loops
	Thu, 10/06	Midterm Exam	
9	**October 9th – 14th**		
	Tue, 10/11	Functions Implementing and Testing Parameters and Arguments, Return Value Boolean Functions (Ch 6 Zy, Ch 3, 6 How)	
	Thu, 10/13	Functions Implementing and Testing Parameters and Arguments, Return Value Boolean Functions	

		(Ch 6 Zy, Ch 3, 6 How)	
10	**October 17th – 21st**		
	Tue, 10/18	More Functions Program Development Problem Solving: Stepwise Refinement Unit Testing Problem Solving: Test Cases (Ch 6 Zy, Ch. 6 How)	Quiz 6. Functions Assignment 7. Functions
	Thu, 10/20	More Functions Program Development Problem Solving: Stepwise Refinement Unit Testing Problem Solving: Test Cases (Ch 6 Zy, Ch. 6 How)	
11	**October 24th – 28th**		
	Tue, 10/25	Logical Errors Debugging with print IDLE Debugger (LN, Ch 6 How)	Assignment 8. Functions. Input Validation Challenge Activities. Chapter 6 Zy
	Thu, 10/27	Strings Processing Strings Analyzing Strings (Ch 7 Zy, Ch 8 How)	
12	**October 31st – November 4th**		
	Tue, 11/01	Strings Processing Strings Analyzing Strings (Ch 7 Zy, Ch 8 How)	
	Thu, 11/03	Modules (random, time, math) Reading and Writing Text Files (Ch 11, 12 Zy, Ch 13 How)	Quiz 7. Strings Assignment 9. Strings Challenge Activities. Chapter 7 Zy
13	**November 7th - 11th**		
	Tue, 11/08	Modules (random, time, math) Reading and Writing Text Files (Ch 11, 12 Zy, Ch 13 How)	
	Thu, 11/10	Lists Lists Properties and Operations List Algorithms (Ch 8 Zy, Ch 11 How)	
14	**November 14th – 18th**		

	Tue, 11/15	Lists Lists Properties and Operations List Algorithms (Ch 8 Zy, Ch 11 How)	
	Thu, 11/17	Lists and Functions Multidimensional Lists Dictionaries Dictionary Methods, Iterating over Dictionaries (Ch 8 Zy, Ch 11, 14 How)	Assignment 10. Lists. List Algorithms
15	**November 21st – 25th**		
	Tue, 11/22 Thu, 11/24	Thanksgiving Break. No classes.	
16	**November 28th – December 2nd**		
	Tue, 11/29	Lists and Functions Multidimensional Lists Dictionaries Dictionary Methods, Iterating over Dictionaries (Ch 8 Zy, Ch 11, 14 How)	
	Thu, 12/01	Recursive Functions (Ch 14 Zy, Ch 18 How) Review for the Final Exam	Quiz 8. Lists and Dictionaries Assignment 11. Multidimensional Lists. Dictionaries.
17	**December 5th – 9th**		
	Tue, 12/06	Study Day	
		Final exam:	Final exam

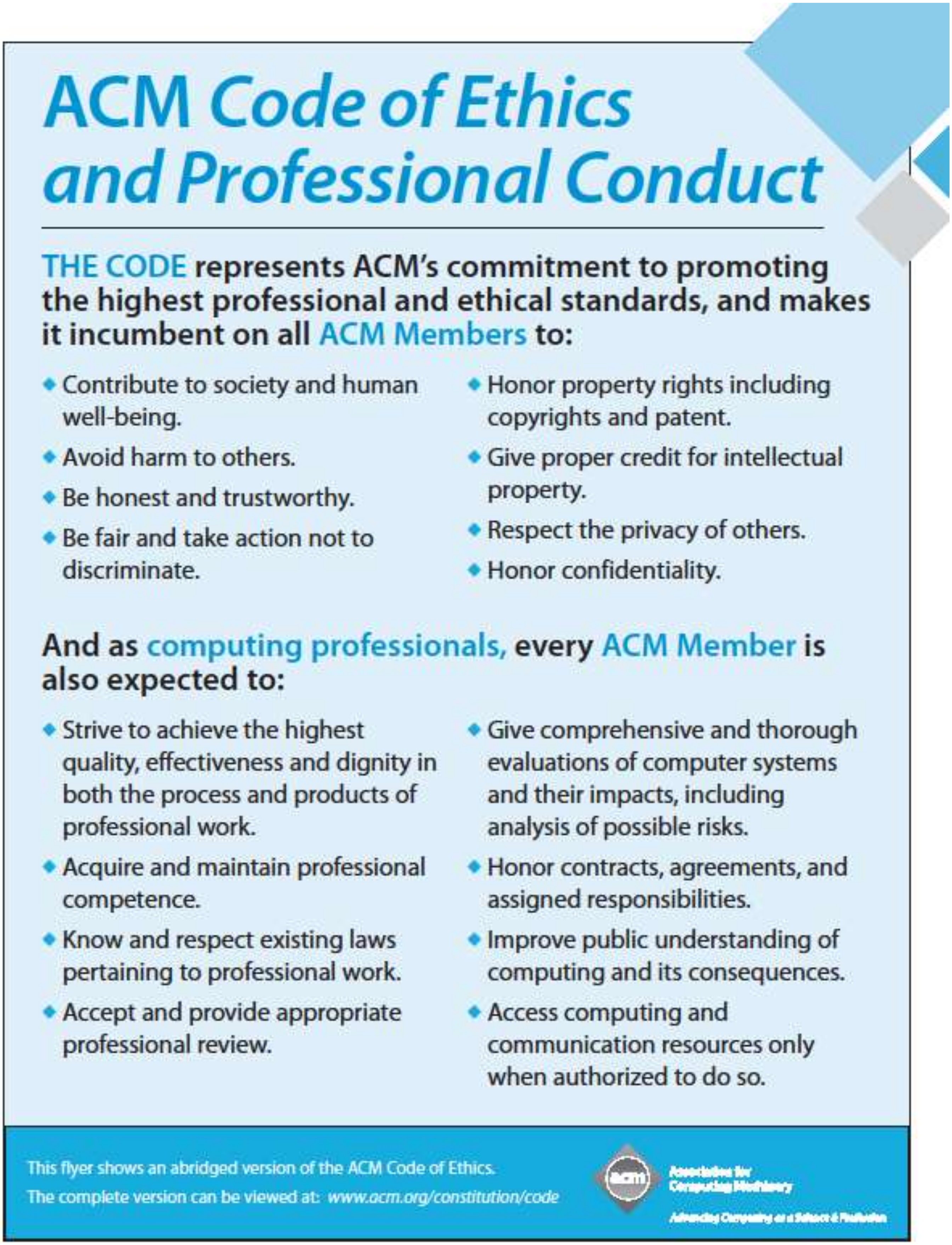
ABET Criteria:
Students in CS/IT will have a(n)
A. ability to apply knowledge of computing and mathematics appropriate to the discipline;
B. ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
C. ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;
D. ability to function effectively on teams to accomplish a common goal;
E. understanding of professional, ethical, legal, security, and social issues and responsibilities;
F. ability to communicate effectively with a range of audiences;
G. ability to analyze the local and global impact of computing on individuals, organizations and society;
H. recognition of the need for, and an ability to engage in, continuing professional development;
I. ability to use current techniques, skills, and tools necessary for computing practice.

J. ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
K. ability to apply design and development principles in the construction of software systems of varying complexity.

CS Program Objectives:

Our graduates will have achieved:

1) A broad general education assuring an adequate foundation in science and mathematics relevant to computing.
2) A solid understanding of concepts fundamental to the discipline of computer science.
3) Good analytic, design, and implementation skills required to formulate and solve computing problems.
4) The ability to function and communicate effectively as ethically and social responsible computer science professionals.



ACM *Code of Ethics and Professional Conduct*

THE CODE represents ACM's commitment to promoting the highest professional and ethical standards, and makes it incumbent on all ACM Members to:

- Contribute to society and human well-being.
- Avoid harm to others.
- Be honest and trustworthy.
- Be fair and take action not to discriminate.
- Honor property rights including copyrights and patent.
- Give proper credit for intellectual property.
- Respect the privacy of others.
- Honor confidentiality.

And as computing professionals, every ACM Member is also expected to:

- Strive to achieve the highest quality, effectiveness and dignity in both the process and products of professional work.
- Acquire and maintain professional competence.
- Know and respect existing laws pertaining to professional work.
- Accept and provide appropriate professional review.
- Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks.
- Honor contracts, agreements, and assigned responsibilities.
- Improve public understanding of computing and its consequences.
- Access computing and communication resources only when authorized to do so.

This flyer shows an abridged version of the ACM Code of Ethics.
The complete version can be viewed at: *www.acm.org/constitution/code*

Association for Computing Machinery
Advancing Computing as a Science & Profession

Please fill this out and upload to the dropbox Student Info on the class CougarView site.

CPSC 1301L (CRN 20533) SPRING 2015

Name:

High School attended:

Current declared major:

Current student status (Freshman, sophomore, junior, senior, postgraduate):

How many course hours are you taking this semester**:

How many hours per week do you work**:

**Military(Y/N)?   Active Duty(Y/N)?   ROTC(Y/N)?   Reserves(Y/N)?

Where can I reach you in case it becomes necessary? **

Email address that you use regularly: ______________________________________

Phone number(s): ____

Do you receive text messages on this phone: _______ YES   _______NO

Do you unlimited received text messages on this phone: _______ YES   _______NO

Declaration: I have read, understood and agree to abide by the policies mentioned in the syllabus pertaining to the course. In particular, I agree to abide by the assignment policy/late work policy, attendance policy, academic dishonesty policy, website policy and exam policy.

(You must sign and date below).

Signature: ________________________________ Date: ________________

** Optional information