COURSE SYLLABUS : CPSC1301 – COMPUTER SCIENCE 1 LAB– FALL 2015
FRESHMAN LEARNING COMMUNITY –SOLVING PROBLEMS USING TECHNOLOGY

INSTRUCTOR INFORMATION

INSTRUCTOR NAME: Dr. Vladimir Zanev
EMAIL: zanev_vladimir@columbusstate.edu
PHONE: (706) 507-8182
OFFICE HOURS AND LOCATION: MWF 10:00 – 11:30 a.m., TR 3:00-4:00 p.m., CCT 442

MEETING TIME AND PLACE:
> CPSC 1301 Summers MWF 8:00-8:50 CCT 405
> CPSC 1301L Zanev MWF 9:00-9:50 CCT 409
> ENGL 1101 Woods MWF 10:00-10:50 HOW 205

COURSE INFORMATION

COURSE CRN NUMBER/TITLE: CPSC 1301L Computer Science 1 Lab (CRN 82224)
CREDIT HOURS/PREREQUISITES: 3 credit hours, co-requisite: CPSC 1301

COURSE DESCRIPTION: This course provides a hands-on experience for the students in the area of programming as a realization of the concepts presented in Computer Science 1. The students will complete programming lab projects using one or more program development environments. The lab projects will apply and reinforce the topics covered in Computer Science 1. (Course fee required.)

REQUIRED TEXTBOOK AND MATERIALS

RECOMMENDED TEXTBOOKS / WEB RESOURCES

http://www.zyBooks.com

E-book
Title: Programming in Python 3.3
Authors: B. Miller, F. Vahid, R. Lysecky
Publisher: zyBooks
Edition: March 2015
http://www.zybooks.com
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

SUPPLEMENTARY MATERIALS

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

LEARNING OUTCOMES

Course Objective: 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 1301L class will meet MWF from 9:00 a.m. to 9:50 a.m. at CCT 409 for three fifty minute sessions each week. The class sessions will focus on problem-solving, Python program development, and Python programming. The class sessions are lab programming oriented. Lab programming assignments have to be developed and completed during the class sessions. The topics of lab programming assignments follow the topics covered in the CPSC 1301 Computer Science 1 course.

Lab Assignments
The lab assignments are individually assignment problem solving, programming assignments. For each assignment, students will be expected to analyze requirements of the problem, design the algorithm and implement the algorithm by coding it into a Python program. The lab assignments will focus on one or more of the learning objectives. The lab assignments have to be completed no later than 11:59 p.m. on the due day. **Late assignments are not accepted for credits**. Lab assignment programs should be submitted through the class CougarView lab assignment dropboxes.

Independent Programming Assignments
The independent programming assignments are short, problem-solving, in-class, Python programming tests. The duration of the independent programming assignments is one class session, 50 minutes. The independent programming assignments have to be completed without any help from the instructor, assistant, or peer during the scheduled class session time. Three independent programming assignments are scheduled.

Final Exam

Your performance in this class will be measured by a comprehensive Final Exam. The Final Exam will be in-class, problem-solving, timed programming exam. During the Final Exam you have to solve problems writing Python programs. **No make-up Final 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 book, online materials and other materials; writing solutions to homework exercises and lab programming assignments.

COURSE EVALUATION

The final grade will be obtained from the following:

Lab Programming Assignments	60%
Independent Programming Assignments	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 (assignments, projects, 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

WEEK	DATE	TOPICS	LAB ASSIGNMENTS, TESTS, FINAL EXAM
1	August 17th – 21st		
	Mon, 08/17	Introduction to the course. Class organization and administration. Problem Solving/Computational Thinking	Introduction to D2L class site, tools, and materials Lab 1. Computational Thinking

	Wed, 08/19	Problem Solving/Computational Thinking	Lab 1. Computational Thinking
	Fri, 08/21	Flowcharts and Algorithmic thinking	Lab 2. Algorithms – Flowcharts and Narrative
2	**August 24th – 28th**		
	Mon, 08/24	Flowcharts and Algorithmic thinking	Lab 2. Algorithms – Flowcharts and Narrative
	Wed, 08/26	Introduction to Computers, Programs, and Python	Lab 3. Introduction to Python and IDLE
	Fri, 08/28	Introduction to Computers, Programs, and Python	Lab 3. Introduction to Python and IDLE
3	**August 31st – September 4th**		
	Mon, 08/31	Elementary Programming: Variables, Arithmetic	Lab 4. Variables, Assignments, and Expressions
	Wed, 09/02	Elementary Programming: First Do It by Hand, Strings	Lab 4. Variables, Assignments, and Expressions
	Fri, 09/04	Elementary Programming: input and output	Lab 5. Input, Output, and Strings
4	**September 7th – 11th**		
	Mon, 09/07	Labor Day. No classes.	
	Wed, 09/09	Simple Turtle Graphics	Lab 6. Simple Turtle Graphics
	Fri, 09/11	Errors Deadline to Withdraw	Lab 7. Errors. Syntax and Runtime Errors
5	**September 14th – 18th**		
	Mon, 09/14	Functions: Introduction	Lab 8. Functions. Introduction
	Wed, 09/16	Functions: Implementing and Testing	Lab 9. Functions. Implementing and Testing
	Fri, 09/18	Functions: Parameters and Returns	Lab 10. Functions. Parameters and Returns
6	**September 21st – 25th**		
	Mon, 09/21	Decisions: Boolean Variables and Operators. The *if* Statement	Lab 11. Decisions. Boolean Variables and Operators. The *if* Statement
	Wed, 09/23	Decisions: Relational Operators, Nested Branches	Lab 12. Decisions. Relational Operators, Nested Branches
	Fri, 09/15	Problem Solving	Independent Programming Assignment 1
7	**September 28th – October 2nd**		
	Mon, 09/28	Functions. Program Development	Lab 13. Program Development

	Wed, 09/30	Functions. Program Development	Lab 13. Program Development
	Fri, 10/02	Input Validation	Lab 14. Input Validation
8	**October 5th – 9th**		
	Mon, 10/05	Input Validation	Lab 14. Input Validation
	Wed, 10/07	Boolean Functions. Unit Testing	Lab 15. Boolean Functions. Unit Testing
	Fri, 10/09	Boolean Functions. Unit Testing	Lab 15. Boolean Functions. Unit Testing
9	**October 12th – 16th**		
	Mon, 10/12	Loops: The *for* Loop	Lab 16. for Loop
	Wed, 10/14	Loops: The *for* Loop	Lab 16. for Loop
	Fri, 10/16	Loops: The *while* Loop	Lab 17. while Loop
10	**October 19th – 23rd**		
	Mon, 10/19	Loops: Hand Tracing, Storyboards	Lab 17. while Loop
	Wed, 10/21	Loops: Nested Loops	Lab 18. Nested Loops
	Fri, 10/23	Loops: tables, other loops	Lab 19. Tables, Other Loops
11	**October 26th – 30th**		
	Mon, 10/26	Strings: Processing Strings	Lab 20. String Processing
	Wed, 10/28	Strings: Analyzing Strings	Lab 21. Analyzing Strings
	Fri, 10/30	Strings: Processing and Analyzing	Independent Programming Assignment 2
12	**November 2nd – 6th**		
	Mon, 11/02	Tuples	Lab 22. Logical Errors. Debugging
	Wed, 11/04	Event-Handling	Lab 23. Event_handling
	Fri, 11/06	Event-Handling	Lab 23. Event-Handling
13	**November 9th - 13th**		
	Mon, 11/09	Lists: Properties and Operations	Lab 24. Lists: Properties and Operations
	Wed, 11/11	Lists: List Algorithms	Lab 25. Lists: List Algorithms
	Fri, 11/13	Lists: List Algorithms	Lab 25. Lists: List Algorithms
14	**November 16th – 20th**		
	Mon, 11/16	Lists: Lists with Functions	Lab 26. Lists with Functions
	Wed, 11/18	Lists: Lists with Functions	Lab 26. Lists with Functions

	Fri, 11/20	Lists: Multidimensional Lists	Lab 27. Multidimensional Lists
15	**November 23rd – 27th**		
	Mon, 11/23 Wed, 11/25 Fri, 11/27	Thanksgiving Break. No classes.	
16	**November30th – December 4th**		
	Mon, 11/30	Modules (random, time, math)	Lab 28. Modules
	Wed, 12/02	Reading and Writing Text Files	Independent Programming Assignment 3
	Fri, 12/04	Reading and Writing Text Files	Lab 29. Reading and Writing Text Files
17	**December 7th – 12th**		
	Mon, 12/07	Recursive Functions Review for the Final Exam	Lab 30. Recursive Functions Review for the Final Exam
	Tue, 12/08	Study Day	Study for comprehensive programming final
		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.

Please fill this out and upload to the dropbox. Also, cut and paste the information into the comment box before you upload it.
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