# CPSC1301L    Computer Science 1 Lab (CRNs 23121, 22161)   Spring 2014

# Class Meets: MWF 10-10:50 AM in CCT 409

###

### **INSTRUCTOR INFORMATION**

### Name: Dr. Yeşem Kurt Peker

### Office: Center for Commerce and Technology (CCT) Room 422

### Office hours:

### Mon: 11–12pm, 2-3pmTue: 11-12pm, 1-2pm

### Wed: 1-4pm

### Thu: 11-12pm, 1-2pm

### Fri: 12:30-1:30pm

### and by appointment

### CATALOG DESCRIPTION OF COURSE

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 projects using one or more program development environments. The lab projects will apply and reinforce the topics covered in Computer Science 1. (1 credit)

*Corequisite*: CPSC 1301

**Required Texts:**

|  |  |
| --- | --- |
| Python3-powered hello-world.svg | Non-Programmer's Tutorial for Python 3URL: <http://en.wikibooks.org/wiki/Non-Programmer%27s_Tutorial_for_Python_3.0>  |

|  |  |
| --- | --- |
|  | [**Big Java: Early Objects, 5th Edition**](http://instructors.coursesmart.com/big-java-early-objects-5th-edition/horstmann-cay-s/dp/9781118431115)**ISBN: 978-1-118-54580-5****AUTHOR:** [**Cay Horstmann**](http://www.horstmann.com/) |

Supplementary Books and Materials

* Materials on course website in CougarView
* Class handouts & notes
* Software and manuals found in the computer labs and on the Internet

**Software** To complete all lessons, assignments, labs, and tests, you will need to access a computer with:

* Windows 7/8/XP/Vista, Linux, or Macintosh, Firefox or Internet Explorer
* Python 3.x
* Java 2 Platform, Standard Edition (J2SE)
* Java IDE (BlueJ/JGrasp/DrJava/Eclipse/ NetBeans/)
* Java Source Code and Program Files from the textbook materials

***Course Material Downloads***

***To download Python (with IDLE):*** <http://www.python.org/download/>

***To download Java SDK:*** <http://java.sun.com/>

***To download Blue J:*** <http://www.bluej.org/download/download.html>

***To download JGrasp:*** <http://www.jgrasp.org/>

***To download Dr. Java:*** <http://www.drjava.org/>
***To download Eclipse:*** <http://www.eclipse.org/downloads/>

***To download NetBeans:*** <http://www.netbeans.org/downloads/>

###### COURSE OBJECTIVE

The aim of this lab-oriented course is to provide hands-on experience with Python and Java programming to reinforce the theoretical knowledge gained in CPSC1301 Computer Science 1.

***Course Learning Outcomes***

 Successful completion of 1301 and 1301L should enable you to:

1. Analyze the requirements of a problem to be solved.
2. Design algorithms to solve moderately complex problems.
3. Work with an Integrated Development Environment (IDE) to
4. Develop Java programs with an editor,
5. Compile a Java program,
6. Run a Java program,
7. Debug Java programs.
8. Develop and use the main Java building blocks: data types, variables, constants, expressions, and arithmetic operators.
9. Use Java classes with constructors, methods, as well as predefined Java classes.
10. Make use of control in programs by means of
	1. conditional statements: *if*, *if/else*, *if/else if*
	2. selection statements: *switch*.
11. Use looping control structures: event-controlled loops (*while* and *do/while* statements), count-controlled loops (*for* statement), and nested loops.
12. Use simple and structured data types (e.g. arrays) to solve problems.

 ***Major Topics***

1. Introduction to the concepts of programming for problem solving
2. Basic elements of Python
3. Data and Expressions
4. Control Structures – Selection
5. Control Structures – Repetition
6. Graphics
7. Input/Output
8. Basic Elements of Java
9. Using Classes and Objects
10. Arrays

###### COURSE OUTCOMES & ASSESSMENT METHODS

* 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.
	+ Program Objectives covered: 2 and 3
	+ Assessment Methods: Programming Assignments.
* The students will demonstrate the ability to design algorithms utilizing the principles of object-oriented programming (classes, encapsulation, inheritance mechanisms, polymorphism) 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.
	+ Program Objectives covered: 3
	+ Assessment Methods: Programming Assignments.
* The students will demonstrate the ability to design algorithms utilizing some principles of programming (exception handling, recursive programming, and basic data structures) 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.
	+ Program Objectives covered: 3
	+ Assessment Methods: Programming Assignments.
* 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.
	+ Program Objectives covered: 3
	+ Assessment Methods: Programming Assignments.
* 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.
	+ Program Objectives covered: 2 and 3
	+ Assessment Methods: Programming Assignments.
* 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.
	+ Program Objectives covered: 2 and 3
	+ Assessment Methods: Programming Assignments.

###### ASSESSMENT CRITERIA

There will be approximately one programming assignment per class. The assignments will have different weights. The maximum score for an assignment will be indicated in the assignment description.  The grading for assignments will work as follows:

1.    You need to be present in class working on the assignment for it to be graded.

2.    You are expected to finish your program by the end of the day and submit your work in the appropriate Dropbox in the course site in CougarView before midnight that day.

3.   I’ll provide you with feedback the next day and you will get a chance to make changes on your assignment according to the feedback and resubmit it until the second class period after the assignment is due.

4.    No late assignments will be accepted.

After your total score is converted to a percentage, letter grades will be assigned according to the following criteria:

 **A (90-100)**: The student fulfills or exceeds all of the assigned content requirements. The student’s knowledge of the subject is accurate throughout. The student exhibits convincing range and quality of knowledge, having done appropriate research, if applicable.

**B (80-89**): The student fulfills all of the important assigned content requirements. The student’s knowledge of the subject is accurate throughout except in minor details. The student seems informed on the subject, having done appropriate research, if applicable

**C (70-79**): The student fulfills most of the important assigned content requirements. The student’s knowledge of the subject is generally accurate, though flawed. The student exhibits limited range or quality of knowledge, having done limited appropriate research, if applicable.

**D (60-69**): The student fulfills some of the important assigned content requirements. The student’s knowledge of the subject is generally accurate, though flawed. The student exhibits limited range or quality of knowledge, having done minimal appropriate research, if applicable.

**F (0-59**): The student fails to address the important requirements of the course.
The student’s knowledge of the subject is generally inaccurate. The student’s knowledge of the subject lacks range or quality

###### ****Instructional Methods and Techniques****

1. The class will meet three times a week for fifty minute lab periods each week.
2. Each student is expected to attend all class lectures, to read the textbook chapters and to make notes. Students will be expected to participate in classroom discussions, both in class and online.
3. Students must have access to computers for doing assignments.
4. 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 programming projects.

**How to Access the Course**

CougarVIEW uses a course management system called Desire2Learn (D2L). For our purposes, the terms CougarVIEW and D2L will mean the same thing. The CougarView Starter Guide is available for your review at [www.columbusstate.edu/cougarviewstarterguide](http://www.columbusstate.edu/cougarviewstarterguide).

This course includes CougarView. You can access CougarView at: https://colstate.view.usg.edu/

Your CougarVIEW username is the same as your CougarNet login ID & password.

Students who add courses during the first week of the semester, should be able to access their newly added course within 24-36 hours.
For additional help contact the CSU Help Desk (CCT building level 1, phone:706-507-8199).

Once you've entered CougarView, you will see a list of courses you have access to. The CPSC 1302 course is listed as "Computer Science 2". Next to this, you should see my name as the instructor. You may also see new calendar postings and email 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.

Note: One common reason for not being able to see the course in CougarView after you log in is late enrolment in the course. From past experience, it usually takes a couple of days after enrolment for the updated student database to be reflected in CougarView.

Once you have clicked on the course's name and accessed the course, you will find a home page with a navigational bar displaying the course title and a set of links that is used to navigate between tools and homepages. Each course and home page has its own navigational bar that links to relevant tools and contents. Contents such as lecture notes and assignments will be progressively added to the course during the semester.

It is your responsibility to frequently look at the course website in CougarView 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.

###### ****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,
* actively participate in discussions,
* attempt all exercises given during the class,
* read any e-mail sent by the instructor and respond accordingly within 48 hours.

*“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:

* introduce the topic to be learned each week and explain concepts and techniques
* clarify topic issues and answer students’ questions,
* post homework assignments in CougarView and introduce them in the class,
* 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 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 Web Server Space**

All currently enrolled CSU students (including online students) can request free Web server space on the CSU student Web server. Simply go to <http://webs.columbusstate.edu/personal/> and click on the link for free Web space.

***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.

CLASS ATTENDANCE: 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. At my discretion, I may drop you from the course for more than ***six(6)*** absences. Missed classes caused by participation in documented, formal, University-sponsored 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. Refer to the CSU Catalog (<http://ace.columbusstate.edu/advising/a.php#AttendancePolicy>) for more information on class attendance and withdrawal.

|  |
| --- |
| **Academic dishonesty**Academic dishonesty includes, but is not limited to, activities such as cheating and plagiarism. It is a basis for disciplinary action. Collaboration is not permitted on assignments or exams/quizzes in this course. 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. The penalty for the first occurrence of academic dishonesty is a zero grade on the assignment or exam/quiz; the penalty for the second occurrence is a failing grade for the course. For exams/quizzes, access to any type of written material or discussion of any kind (except with me) is not allowed. (<http://ace.columbusstate.edu/advising/a.php#AcademicDishonestyAcademicMisconduct>)**Getting help** Student assistants at the Helpdesk in level 1 of the CCT building 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 at the School of Computer Science lab (CCT450) who can help you with the assignments. Their schedule is posted in the Computer Science department. You can always contact me during my posted office hours, by e-mail, or by appointment. |

**Electronic Devices and Academic Integrity**: All cell phones 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, and laptops. Any use of such a device during a test or quiz will be considered a breach of academic integrity.

**CSU ADA statement**
"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. 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."

***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

###### COURSE OUTLINE (tentative)

|  |  |
| --- | --- |
| **Week** | **Topics** |
| Week 1: (1/13-1/17) | Introduction to programming and computersIntroduction to Python |
| Week 2: (1/20-1/24) | No class on Monday (MLK Day)Python: Variables and manipulating themGetting input from the userIntroduction to Loops |
| Week 3: (1/27-1/31) | Python: More loopsDecisions (if statements) |
| Week 4: (2/3-2/7) | Python: FunctionsLists |
| **Monday, February 3 - LAST DATE TO WITHDRAW WITH A ‘W’** |
| Week 5: (2/10-2/14) | Python: For loopsBoolean expresiionsMore lists |
| Week 6: (2/17-2/21) | Introduction to JavaJava: Types and variables |
| Week 7: (2/24-2/28) | Java: Objects, classes, methods |
| **Spring Break (no classes) – March 3-7** |
| Week 8: (3/10-2/14) | Java: Objects, classes, methods |
| Week 9: (3/17-3/21) | Java:Using classes- Graphics applications  |
| Week 10: (3/24-3/28) | Java:Implementing classes |
| Week 11: (3/31-4/4) | Java:Implementing classes |
| Week 12: (4/7-4/11) | Java: Data types, arithmetics, Strings, Input /Output |
| Week 13: (4/14-4/18) | Java: If statements, Boolean expressions,Multiple alternatives – switch statement |
| Week 14: (4/21-4/25) | Java: While loopDo-While loop |
| Week 15: (4/28-5/2) | Java: For loopArraysEnhanced for loop |
| Week 16: (5/5-5/9)  | Class on Monday - REVIEW, Study Day on May 6  |

**Please return the following information to me as soon as possible.**

**CPSC 1301 (CRNs 23121 and 22161) Spring 2014**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (please print)

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

 Email address that you use regularly: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 Phone number(s): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

 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

<http://computer.howstuffworks.com/computers-videos-playlist.htm>