

CPS121A - INTRODUCTION TO PROGRAMMING

Professor and Lab L1:	Russell C. Bjork russell.bjork@gordon.edu ;	Fall Semester, 2017
Lab L2:	Russ Tuck russ.tuck@gordon.edu	
Office:	KOSC 242 x4377	MWF 1-2 pm Chase B30
Hours:	MWF 3:20-4:20 pm; Tu 1:30-4:30 pm and by appointment	Lab L1: T 9:45am-12:45 pm Lab L2: T 1:15-4:15 pm Both in KOSC 118
Course Site:	Blackboard sites for CPS121A + CPS121L1 (for both labs)	

CATALOG DESCRIPTION:

Introduces fundamental programming concepts using Python. Topics include nature of algorithms, essential control structures, data representation, and programming paradigms. Weekly programming laboratories.

COURSE OBJECTIVES:

There are literally hundreds of programming languages. One comprehensive list starts at A+, wanders through languages with names like C#, Scheme, and Tcl, and finally ends with ZPL. In fact, new languages are developed all the time. However, there are some common structures, which bind a majority of these languages together. In this course, the student will be given the opportunity to learn and understand these essential components of programming.

In particular, upon completion of this course, you should be able to:

1. Think algorithmically.
2. Develop various algorithms,
3. Express simple algorithms as programs using the basic capabilities of the Python language.
4. Perform simple algorithmic manipulations on picture, sound, text and movie files using Python.
5. Manipulate information represented in html or xml using Python.

TEXTS: Guzdial and Ericson. *Introduction to Computing and Programming in Python, A Multimedia Approach* (4th ed) (Upper Saddle River, NJ: Prentice Hall, 2016) ISBN: 978-0-13-402554-4

Elkner, Downey, and Myers *How to Think Like a Computer Scientist: Learning with Python* (2nd ed) <http://openbookproject.net/thinkcs/python/english2e/>

ONLINE TUTORIALS: w3Schools.com. *HTML Tutorial*.
<http://www.w3schools.com/html/default.asp>
w3Schools.com, *Basic XML Tutorial*.
<http://www.w3schools.com/xml/default.asp>

PREREQUISITES:

While no formal prerequisite coursework is required before attending this course, it is important for all students to be aware that this class has components which require strong mathematical reasoning, logical reasoning, and abstract reasoning. Consequently, students with weak ability in

one or more of these areas are strongly encouraged to discuss this with the professor during the first week of classes to determine whether the course is suitable for their needs.

COURSE TECHNIQUES AND PROCEDURES

Since this course is primarily concerned with the development of certain skills and habits, regular practice with evaluation will be the heart of the course. For most units of material, you will be asked to read a portion of one or both of the text books or an online tutorial.

Class sessions will include a discussion and amplification of the material in the text and the presentation of further examples and supplementary material. You should not expect to grasp everything presented in the text when you first read it; however, you should note areas that are unclear to you and be prepared to raise questions about them in class.

As is true with any skill, the only way you can really learn programming is by doing it. For this reason, you will have opportunities to practice what you are learning through weekly laboratories and several programming projects.

COURSE REQUIREMENTS AND EVALUATION:

The college has adopted the following statement regarding work expectations for courses; "For each semester hour of credit, students should expect to spend a minimum of 2-3 hours per week outside of class in engaged academic time. This time includes reading, writing, studying, completing assignments, lab work, or group projects, among other activities."

1. You will be expected to read most of the Guzdial text, and selected chapters from the Elkner text, as assigned in the schedule below. Reading assignments should be completed **BEFORE** the class hour in which the topic is discussed, as specified in the schedule below, and there may be a short reading quiz on the days reading is due. (Reading quizzes will be worth a total of 3% of the final course grade; the lowest three scores will be discarded. Missed quizzes cannot be made up. If you are late for class on the day a reading quiz is given, you will miss the quiz.) However, our classroom discussion will not rigidly follow the order of material in the text, nor will it be confined to material covered there.
2. The chapters in Guzdial include a set of Problems at the end. Though these will not be formally assigned, we will do a number of them in class sessions. For this reason, you should be sure to bring your book with you to class when we are discussing topics in it.
3. Weekly laboratories will focus on gaining practical experience with the material covered in the book and/or in lecture. Lab assignments will be posted on Blackboard (on the CPS121 L1 site) ahead of time, and **must** be read over carefully **before** coming to lab. In some cases, you will be explicitly directed to study certain material in preparation for the lab. For most laboratories, there will be a writeup to turn in. There may also be a quiz given at the start of the lab hour (based on your reading of the lab assignment and any assigned pre-lab preparation) and/or a quiz based on the work done in lab given when you complete the lab. Each lab with a formal writeup/quizz(es) will account for 2% of the course grade (24% total for twelve such labs). The tentative lab emphases are shown in the course schedule.
4. Four programming projects will be posted on Blackboard. These projects must be done in accordance with the handout "Guidelines for Computer Science Projects", which will be distributed with the first project. You are expected to read these carefully and comply with them exactly. The first three projects will each be worth 8% of the final course grade and the last project will be worth 12%. (36% total for the 4 projects).

5. A mid-term examination and a final examination (worth 17% and 20% of the final course grade, respectively) will be given as shown in the course schedule. Each exam will assume familiarity with material in the text, covered in lecture, and/or used in labs or projects. Exams will be open book (course text only), open notes.
6. Your final grade will be computed on the basis of a weighted sum of the items listed above.

Summary:	Reading Quizzes	3%
	Labs	24%
	Projects	36%
	Exams	<u>37%</u>
		100%

The following are minimum guaranteed grades for the percentages indicated:

	93% - 100%: A	90% - 92.9%: A-
87% - 89.9%: B+	83% - 86.9%: B	80% - 82.9%: B-
77% - 79.9%: C+	73% - 76.9%: C	70% - 72.9%: C-
67% - 69.9%: D+	63% - 66.9%: D	60% - 62.9%: D-

ACADEMIC DISHONESTY

From the Gordon College Student Handbook: "Academic dishonesty is regarded as a major violation of both the academic and spiritual principles of this community and may result in a failing grade or suspension. Academic dishonesty includes plagiarism, cheating (whether in or out of the classroom) and abuse or misuse of library materials when such abuse or misuse can be related to course requirements." For the purposes of this course, abuse or misuse of Gordon computer systems or networks related to course requirements will also be viewed as academic dishonesty.

Academic dishonesty will not be tolerated. You know better. Just don't!

COMPUTER USE IN CLASS:

If you have a laptop, please bring it to class. However, electronic devices (computers, tablets, phones, etc.) may be used in class ONLY for class work, including note taking, consulting class materials, and in-class assignments. Using electronic devices for non-academic activities during class (email, social networking, games, internet browsing, etc.) will cause the student to lose the privilege of using electronic devices in class.

POLICY STATEMENT ON EXTENSIONS AND INCOMPLETES:

1. Extensions of the due dates for homework or projects MAY be given in the event of extenuating circumstances (such as illness, personal emergency) IF you submit a brief written request to the professor as soon as possible after the circumstances arise.
2. A grade of Incomplete MAY be given without penalty IF you are unable to complete the course work by the last day of the term due to major illness or other similar emergency. You must apply for this using the form provided by the registrar. Such a request will only be granted if you are substantially up-to-date with your course work and were making good progress in the course up to the time that the difficulty arose. Of course, you must complete all work for the course by the midpoint of the next semester in accordance with College policy.

ATTENDANCE POLICY:

Regular class attendance is expected of all students, and class attendance will be recorded. Absences from class will be classified as “documented” or “undocumented”. A documented absence is one where written documentation is submitted supporting an absence from class due to circumstances beyond the student’s control. An undocumented absence is any other absence, including one which could qualify as documented if proper documentation were submitted.

Students who have more than three undocumented absences during the semester should expect to see their final grade reduced by 1% for each undocumented absence over 3, and students who have more than 12 undocumented absences will fail the course automatically. The allowance of 3 undocumented absences will be reduced by one for each documented absence over 3 - e.g. a student who has 5 documented absences will be allowed only 1 undocumented absence without grade penalty. (This will not be applied retroactively, though) Note that it is not necessary to document absences unless there are more than three total absences; for most students, this will avoid the need to submit documentation. A student who anticipates the need to miss more than three classes due to athletic competitions or other student activities should review the college’s attendance policy in the catalog, and should then discuss alternatives to class attendance with the professor at the start of the semester.

A student who is habitually late will have late arrival for class counted as a half absence for that class, and a student who sleeps through most or all of a given class session will be counted as absent for that class.

You may ask the professor to waive this policy for you if you earned an A in the prerequisite course, or if you have an A average in this course as of the mid-term exam. If you wish to take advantage of this exemption, you must so inform the professor. However, the attendance policy will be reimposed if your subsequent work deteriorates.

STUDENTS WITH DISABILITIES:

Gordon College is committed to assisting students with documented disabilities (see Academic Catalog Appendix C, for documentation guidelines). A student with a disability who may need academic accommodations should follow this procedure:

1. Meet with a staff person from the Academic Support Center (Jenks 412 X4746) to:
 - a. make sure documentation of your disability is on file in the ASC,
 - b. discuss the accommodations for which you are eligible,
 - c. discuss the procedures for obtaining the accommodations, and
 - d. obtain a **Faculty Notification Form**.
2. Deliver a Faculty Notification Form to each course professor *within the first full week of the semester*; at that time make an appointment to discuss your needs with each professor.

Failure to register in time with your professor and the ASC may compromise our ability to provide the accommodations. Questions or disputes about accommodations should be immediately referred to the Academic Support Center. (See also Grievance Procedures in Student Handbook)

TENTATIVE COURSE SCHEDULE - SUBJECT TO REVISION

<u>Date</u>	<u>Topic(s)</u>	<u>Reading</u>	<u>Written Work Due</u>
W 8/30	Course Introduction		
F 9/1	Introduction to Computer Science and Media Computation	Guzdial ch. 1	
M 9/4	<i>Labor Day - no class</i>		
T 9/5	Lab: JES and Python (1)		
W 9/6	Introduction to Programming	Guzdial ch. 2; Elkner ch. 1	
F 9/8	Modifying Pictures Using Loops	Guzdial ch. 4	
M 9/11	(continued)		
T 9/12	Lab: Modifying Pictures 1 (2)		
W 9/13	Using Selection with Pictures	Guzdial ch. 5	
F 9/15	Using Selection (continued)		
M 9/18	Modifying Pixels by Position	Guzdial ch. 6	
T 9/19	Lab: Modifying Pictures 2 (3)		
W 9/20	Modifying Pixels by Position (continued)		(Start Project 1)
F 9/22	Modifying Sounds Using Loops	Guzdial ch. 7	
M 9/25	Modifying Samples in a Range	Guzdial ch. 8	
T 9/26	Lab: Modifying Pictures 3 (4)		
W 9/27	Combining Sounds	Guzdial ch. 9	
F 9/29	Arrays and Lists	Elkner § 9.1-9.9	
M 10/2	Application of Arrays and Lists; Modeling and Simulation		
T 10/3	Lab: Manipulating Sounds (5)		
W 10/4	Creating and Modifying Text	Guzdial ch. 3; Elkner ch. 7	
F 10/6	(continued)		
M 10/9	Files and Text	Guzdial ch. 11	PROJECT 1 DUE - Start Project 2
T 10/10	Lab: Strings (6)		
W 10/11	Files and Text (continued)	Elkner §10.7-10.9	(Start Project 2)
F 10/13	Variables, Expressions, and Statements	Elkner ch. 2	
M 10/16	Review and Catch up		
T 10/17	Lab: Files and Text (7)		
W 10/18	MIDTERM EXAM		(MIDTERM EXAM)
F 10/20	<i>Quad Break - no class</i>		
M 10/23	Functions	Elkner ch. 3	
T 10/24	Lab: List Operations (8)		
W 10/25	Accessing Data from the Web	Guzdial ch. 12	
F 10/27	Making Text for the Web; HTML	Guzdial ch. 13	
M 10/30	(continued)	Look over HTML Tutorial	PROJECT 2 DUE

T 10/31	Lab: Creating HTML (9)		
W 11/1	(HTML continued)		(Start Project 3)
F 11/3	XML	Look over XML Tutorial	
M 11/6	(continued)		
T 11/7	<i>Day of Prayer - no lab</i>		
W 11/8	XML (continued)		
F 11/10	(continued)		
M 11/13	Building Bigger Programs; Test-First Development	Guzdial ch. 10	
T 11/14	Lab: XML (10)		
W 11/15	Boolean Expressions; More about Conditionals	Elkner §4.2-4.7	
F 11/17	Iteration	Elkner §6.3-6.6	
M 11/20	Tuples and Dictionaries	Elkner § 11.1-11.3	PROJECT 3 DUE;
		Elkner § 12.1-12.4	(Start Project 4)
T 11/21	Lab: Unit Testing (11)		
W 11/22	<i>Thanksgiving Break - no class</i>		
F 11/24	<i>Thanksgiving Break - no class</i>		
M 11/27	Programming Paradigms		
T 11/28	Lab: Project Work Session (optional if part I of project is turned in before lab)		PROJECT 4 PART I DUE AT THE END OF LAB
W 11/29	The Functional Paradigm; Recursion	Guzdial ch. 16; Elkner § 11.6, 11.8, 11.9	
F 12/1	(Functional / Recursion continued)		
M 12/4	The Object-Oriented Paradigm	Guzdial ch. 17	
T 12/5	Lab: Functional Programming (12)		
W 12/6	Searching and Sorting		
F 12/8	Searching and Sorting (continued)		
M 12/11	Movies	Guzdial ch. 14	
T 12/12	Lab: Project Work Session (optional if Part II of project is turned in before lab)		
W 12/13	Review and Catch up		PROJECT 4 PART II DUE