

Course Syllabus

Course Title: Programming Logic & Technique, CIS1400-VCM01 FL 2020

Credit hours: 4 -Clinical Hours: 0 Lecture Hours: 4 Lab Hours: 0

Instructor: Robert Burrows

E-Mail: burrows@cod.edu - will check email at least every other day during semester

Office Hours/additional help: By request – there will be periodic help sessions arranged outside of class as requested by students

Classroom, Day & Time: Virtual Course - Tuesday, Thursday 2:00 – 3:50 PM

Class Dates: 08/25/2019 – 12/17/2019

Textbook & Materials:

Starting Out with Programming Logic and Design, 5th Ed. By Tony Gaddis, 2019 ISBN: 978-0-13-480115-5, Publisher: Pearson.

-Appendix E to book has answers to Checkpoint questions within each chapter that are a good review of reading.

- I will supply answers to the review questions at the end of each chapter. Doing the review questions is an excellent gauge if you understand the chapter's lessons

- At the end of each chapter there are Python examples that match the book's program examples (Book uses pseudocode – English like description of program- or flowchart to describe program logic in the chapter. There are also Java, C++ examples at the end of each chapter in the book. Though a couple of references to Java and C++ in class, perhaps you are interested in these languages as your next programming class and may want to look at them for each chapter; but please only after completing your Python work and looking at the Python examples ☺).

Follet Discovery link, <https://www.bkstr.com/dupagestore/home/en> to buy or rent the textbook from the campus bookstore (will actually look up course(s) and books needed for each course by your student id). Online you may find an online version of the book that is cheaper if you don't need paper or hardcopy.

We will use the Python programming language in this class. Python is a relatively easy language compared to Java and C++.

You will need a PC or MAC for the class. The larger the monitor you have for online classwork the better, however I will tailor the online instruction so that a 13" laptop will be sufficient. If you don't have a computer and cannot afford one at this time the college will lend you one to use.

We will install the Python language on your Mac or PC in the first class from:

<https://www.python.org/downloads/>.

To be successful in class, besides a computer and the text, you will need a good internet connection for the online classroom lectures. See the first announcement in Blackboard to test your internet connection.

Course Description

An introduction to computer-based problem-solving techniques. Includes software design tools such as structure charts, Input Processing Output (IPO) charts, flowcharts, pseudocode, and Unified Modeling Language (UML) diagrams. Concepts such as documentation, structured design, modularity, Object Oriented Program (OOP) design, and event driven programming are covered. Programming of algorithms are implemented using a high level language that emphasize structured and object oriented design techniques.

Repeatable for credit: No

Pre-Enrollment Criteria:

Prerequisite: MATH 0482 Foundations for College Mathematics II with a grade of "C" or better, or equivalent or

Prerequisite: MATH 1115 Technical Mathematics I with a grade of "C" or better, or equivalent or or a qualifying score on the mathematics placement test or

Consent of Instructor

Topical Outline:

1. Program Development Lifecycle (PDLC)
2. Software design techniques
3. Computer based paradigms
4. Simple data types
5. Variables, constants, and literals
6. Control structures
 - a. Sequential
 - b. Selection
 - c. Repetition
7. Arithmetic, relational, and logical operators
8. Local and global variable scope
9. Modularity, parameter passing, and return values
10. File access
11. Advanced data types: Arrays
 - a. Single dimensional
 - b. Multi-dimensional
 - c. Parallel
12. Searching and sorting algorithms
13. Programming and debugging
14. OOP design concepts
15. GUI and event driven programming

General Course Objectives:

Upon successful completion of the course the student should be able to do the following:

1. Explain steps used in program development cycle
2. Identify tools used in software design
3. Create algorithms to solve both verbal and written problems
4. Differentiate simple data types
5. Differentiate variables, constants, and literals
6. Apply concepts of structured program design such as modularity, sequence, selection, and repetition
7. Differentiate arithmetic, relational, and logical operators in algorithm design
8. Demonstrate variable scoping in program design for local and global variables

9. Apply data transfer techniques between modules using parameters and return values
10. Construct applications to use files for input and output
11. Implement arrays as structures to contain data
12. Use searching and sorting algorithms in problem designs
13. Use a higher level programming language to code, test, and debug software designs
14. Implement concepts of abstraction and encapsulation using Object Oriented Programming (OOP) design
15. Explain advanced OOP design techniques such as inheritance and polymorphism
16. Describe integration of Graphical User Interfaces (GUIs) and event driven programming

Course Requirements

Student Responsibilities: Class attendance, active participation, and keeping up with the homework assignments are essential if students are to receive maximum benefit from the class. Participation requires preparation including completion of reading, assignments, assessments and exams by the due dates. All due dates will be listed on the class blackboard. Students are responsible to submit all work on or before the due dates. Questions, comments and suggestions are welcome.

Weekly Quizzes on reading must be submitted by the due date or you will receive 0 points

Exams and make-up policy: There will be four exams. No retakes of exams are allowed. Make-up exams are not encouraged except on emergency situation.

Assignments (including Labs, Assessment): It is extremely critical that students complete all assignments on time. Past experience has shown that students that were behind on their assignments never caught up. Students may submit assignments early. Submitting assignments in the order assigned will ensure progression according to academic design of the course. If an assignment is late you will get only half credit.

NOTE do to Covid-19 I am very understanding, especially for first responders. Let me know if you need extra time for any quizzes or assignments and I will grant an extension

E-mail: Every attempt will be made to answer e-mail on a daily basis. When sending e-mails please indicate clearly the problem or concern you are having, your name and course enrolled. All communications and e-mail will be through the COD email and blackboard.

Academic policy: Any violation of COD policies regarding academic honesty and/or integrity will be referred automatically to the appropriate college authorities for disposition. Please see appropriate pages in the college catalog for definitions and regulations. The minimum penalty for cheating will be a zero for all parties involved on that exam, assignment, lab, or assessment.

Withdrawal policy: Missing three or more assignments, assessments and or labs before mid-term will result in the student withdrawn from the class for Non-Pursuit of the course objectives. No longer attending a class does not constitute an automatic withdrawal. **All withdrawals must be done on or before Sunday November 15th** as in the College's academic calendar (<https://cod.edu/academics/calendar.aspx>) After that date you will no longer be allowed to drop the course.

Class Behavior and set up: This is an internet class so an internet access should be set up at least 5 minutes before class starts. We will be using collaborate ultra of blackboard. During class time, considerate conduct by all persons is important to a favorable learning environment. Any infringement on the rights of others to get education will be dealt with in an appropriate manner. Please try to have no distractions during class and have your computer on mute unless asking a question. General notes: In order to achieve the course objectives, it is

essential that you enjoy the class in addition to complying with the above requirements, the rules and policies of COD contained in the catalog and other COD materials. Most students sign up for courses with the best intentions. If you are experiencing course/college related problems, please feel free to talk to the instructor before a crisis develops so that we can resolve them in a manner beneficial to all parties involved.

Class period rough outline

2:00 – 2:05 Questions from last class

2:05 – 3:00 Lecture/discussion

3:00 – 3:05 Break

3:05 – 3:50 Lecture/discussion/assignment help

3:50 – 4:30 Optional after class Homework/question help session as required

Grading and Evaluation Criteria

Your final grade is based on accumulation points of all class work including homework, quizzes and exams.

Points will be distributed in the following manner:

Programs/Assignments 20 x 10 points/each and 5 x 20	300
Quizzes 13 x 10 points each	130
Tests	120
Extra Credit – up to 40 points	40
Total	590

NOTE: although you can earn 590 points your grade is based on 550 points. Final Grades will be assigned using the following scale which follows 90% > A 80% > B....– **NOTE keep track of your points especially near the end of the term to do extra credit up to 40 points if needed if on the borderline of a grade:**

Accumulated Points	Grade	Percentage
495 – 590	A	>= 90 %
440 – 494	B	80 – 89
385 – 439	C	70 – 79
330 – 384	D	60 – 69
329 or lower	F	< 60 %


*** **NOTE** an **Incomplete grade** is **not allowed** in this class and drop date is November 16th

Tentative Course Outline

Class Dates	Topics	Homework and Quizzes Book chapter actively read prior to class	Due Dates Homework and quizzes
1 Aug 25	Course introduction Programming and algorithms OTTO – types of programming logic Review syllabus Install Python Class student Survey to determine your experience objectives, and motivation for the course.	HW 1: Student Survey in class – 10 points	
2 Aug 27	Textbook Active Reading Chapter 1 Introduction- a look inside the computer	Chapter 1 Book HW 2: Prog <u>Exercise</u> 1-3 your name in ASCII Code 10 points. Note – Ex 1-3 is at the very end of the chapter 1 under Exercises not <u>Program 1-3 example in chapter</u> nor <u>Checkpoint exercise 1-3</u> within chapter	
3 Sept 1	Chapter 2 Input, Process, Output - variables Pseudocode Flowcharts Example: Prog EX 2-8 IPO and pseudocode and flowchart in class	Chapter 2 Book HW 3: Prog EX 2-6 pseudocode and flowchart – 10 points Quiz Chapter 2	HW 1
4 Sep 3	Chapter 2 more examples, review and questions Assignment input and output statements Python language for a simple Program.	Chapter 2, finish up and review	HW 2
5 Sep 8	Chapter 2 finish up Chapter 4 If statements - Decision Structure and Boolean Logic	Chapter 2, finish up and review HW 4 CodeLab chapter 2 – 20 points Chapter 4	Quiz Ch 1
6 Sep 10	Chapter 4 . Example: Program Ex 4-2, 4-5 pseudocode and flowchart in class	HW 5: CodeLab Decision Logic – 20 pts HW 6: Prog EX 4-6 flowchart and Python Quiz Chapter 4 flowchart and Python	Quiz Ch 2
7 Sep 15	Finish Chapter 4 Another decision logic flowchart example Chapter 5 Repetition Structure (looping). finding a parking spot	Chapters 5 HW 7: CodeLab Repetition/loop logic 20 points	HW 3
8 Sep 17	Chapter 5 Repetition Structure (looping). pseudocode algorithm - find a parking spot Prog EX 5-3,5-5 in class Flowchart and Python Pub Crawl flowchart. Review Your Point Total	Quiz chapter 5 HW 8: Prog EX 5-9	Quiz Ch 4 HW 4

The instructor reserves the right to modify, change, or waive any part of the syllabus or the evaluation criteria for this course. The instructor will provide prior notification of any modifications. The class schedule provides a general guideline for planning and preparation purposes only. Actual dates may vary at the instructor's discretion reflecting subject difficulty, length of discussions, addition of supplemental materials, etc.

9 Sep 22	Finish up chapter 5 Chapter 3 Modules – Introduction Divide and Conquer – even though haven't done big programs Prog Ex 3-2	Chapter 3	HW 5 HW 6
10 Sep 24	Chapter 3 Modules – local variables to a module Prog EX 3-5 pseudo code and Python in class	HW 9: Prog EX 3-10	HW 7 Quiz Ch 5
11 Sep 29	Finish up Chapter 3 Chapter 6 Functions	Chapter 6 functions HW 10 – CodeLab functions – 20 points Quiz Chapter 6	HW 8 Quiz Ch 3
12 Oct 1	Chapter 6 Functions passing variables In class design game of Craps – a true need for a flowchart to see logic Built in functions Turtle package Review Your Point Total	Chapter 6 functions HW 11: Prog EX, 6-7	HW 9
13 Oct 6	Chapter 6	HW 12: Prog EX 6-12 (Rock-Paper-Scissors)	HW 10 Quiz Ch 6
14 Oct 8	Chapter 6 finish up and review Chapter 7 Input validation – only 13 pages Prog EX 7-5 in class – edit Rock, Paper, Scissors Practice Midterm – handout to try chap 1-7	Chapter 7 – (only 10 pages) HW 13 Prog EX 7-1 No Quiz for chapter 7	HW 11
15 Oct 13	Review practice midterm	Review – Home catch up and questions	HW 12
16 Oct 15	Midterm chapters 1-7 - 50 points Review Your Point Total		HW 13
17 Oct 20	Arrays Go over Midterm Chapter 8 Array/Lists and Flowcharting. Goes with Repetition – loops and arrays like pizza and beer or pizza and pop.	Chapter 8 Arrays Quiz Chapter 8 HW 14 CodeLab arrays/lists 20 points	
18 Oct 22	More Arrays Prog EX 8-3 in class parallel arrays	HW 15 Prog EX 8-2 (write lottery number horizontally) Extra Credit Tic Tac Toe (I will provide part of program and go over in class) and (Lo Shu Magic Square) 5 points extra credit each	
19 Oct 27	Chapter 9 Sorting and Searching Arrays Swapping 2 array elements	Chapter 9 sorting/searching Quiz Chapter 9	HW 14 Quiz Ch 8
20 Oct 29	Finish up Chapter 9 and review	HW 16 Prog EX 9-2(only use 5 names not 20 as problem states) in class HW 17 Prog EX 9-4 use/copy bubble sort and binary search code given in class examples	HW 15

21 Nov 3	Chapter 10 Files Prog Ex 10-1 Examples of Python programs with sequential ASCII files	Chapter 10 Quiz Chapter 10	Quiz Ch 9
22 Nov 5	Finish up Chapter 10 and review Review point total	HW 18, Files CodeLab 10 points HW 19 Prog EX 10-2 (just print count of names in the file) also given data file with assignment	HW 16 HW 17
23 Nov 10	Chapter 11 Menu Driven programming and debugging Example 11-4 and Book example on Metric Conversion NOV 15th last day to withdraw	Chapter 11 Quiz Chapter 11 HW 20: Prog EX 11-1 (Language Translator)	Quiz Ch 10 HW 18
24 Nov 12	Chap 12 Text Processing Example Prog EX 12-2 scan for periods. DO A FLOWCHART	Chapter 12 HW 21 Strings CodeLab 10 points HW 22: Prog EX 12-1 backward string in class – can get 3 points extra if you use 4 different algorithms – hint: google how to reverse string in Python Quiz Chapter 12	HW 19
25 Nov 17	Chap 14 OOP Pets example Lynda Video on OOP	Chapter 14 Object Oriented Programming Quiz Chapter 14	Quiz Ch 11 HW 20
26 Nov 19	Chap 14 OOP and review Prog EX: 14-1 in class	HW 23 OOP CodeLab 10 points	Quiz Ch 12 HW 21
27 Nov 24	Chap 14 Finish	HW 24 Prog Ex 14-2	Quiz Ch 14 HW 22
NOV 26	No class happy Thanksgiving		
28 Dec 1	Perhaps bit more on OOP Chapter 15 GUIs Review point total GUI Example of Metric conversion program	Chapter 15 GUI Quiz Chapter 15	HW 23
29 Dec 3	Chapter 15 GUI and review HW 25 Prog EX: 15-1 in class	HW 25 Prog EX: 15-3	HW 24
30 Dec 8	Finish up chap 15 Practice Final Exam handed out		Quiz Ch 15
31 Dec 10	Practice Final Exam review for final Help to complete all homework but should have almost all done by now!	Homework help and Questions on class for final	HW 25
32 Dec 15	Final Exam	All work due 11:59 PM 12/17	All work in 12/17

***NOTE:** Assignments, Assessments, and hands-on labs will be available on the class blackboard.*

Course Expectations

1. All assignments, (including assessments, projects, labs, and any related course work) must be submitted on or before the given due dates for full credit. **Late homework assignments and quizzes start at half credit.**
2. Late work may be given full credit based on the situation with the instructor's consent.
3. ~~When an assignment due date has passed, it will disappear from the class Blackboard (Bb). Please do not ask the instructor to reset the assignment due date or inform the instructor that you can no longer see the assignment on the Bb.~~ Due to Covid 19 this is rescinded
4. To receive full points, a complete assignment must be submitted by following the assignment's requirements and specifications.
5. Unless requested by the instructor, no assignments should be sent to the instructor's email address. Any assignments sent to the instructor's email without permission will be disregarded.
6. All assignments must be submitted on the class Bb. For programming courses, all programs source code listings must be presented with the program output/result.
7. Submitting assignments in the order assigned will ensure progression according to the academic design of the course.
8. Discussions and collaborations are permitted. However, you must do your own assignments. Dishonest work will be disciplined according to the university's policy.
9. If you are caught copying someone's work, you will be placed in one of the following possibilities (depending on the work):
 - a. You get a zero for your current assignment.
 - b. You will get one letter grade less in the course; i.e. if you get an A, you will get a B grade.
 - c. You will be expelled from the university.If you are caught copying above, all of your previous work relating to the course will be re-examined and re-evaluated. All of your future work in the course will be closely scrutinized.
10. You understand the grading criteria as mentioned in the syllabus. The grading distribution will not be curved.
11. Your attitude and behavior relating to the course will be considered when a borderline grade befalls to boost up your grade to the next higher letter grade.
12. An incomplete grade is allowed only in last two weeks of class due to an unforeseen event. Being behind on homework is not an unforeseen event.
13. You are responsible for withdrawing from the course if no longer attend the class. Please check the university's academic calendar on the last day allowed to withdraw from the class.
14. **Bad planning on your part is not an emergency on the instructor's part.**