

Course Syllabus

Course Title: Programming Logic & Technique, CIS1400-004 SP 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 via email– there will be periodic help sessions arranged outside of class as requested by students

Dates, Day & Time, Classroom: 02/18/2020-05/14/2020 Lecture/Discussion Tuesday, Thursday 02:00PM - 04:25PM, Seaton Computing Center(SCC), Room 115

Textbook & Materials:

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

-Appendix to book has answers to checkpoint questions at end of each chapter that are a good review of reading.

- 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)

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

You will have Python on the computers in the classroom. If you want to use your own PC, MAC, laptop or MacBook, you can download the Python language, also free from <https://www.python.org/downloads/> Select version 3.7 or higher. I will go over this in class and also how you can set up windows to bring up python when you click on a python program like windows brings up Microsoft Word when you click on a .docx file.

To be successful in class, you will need your own computer for home use to do assignments.

Finally, you can use either a PC or Mac for this class, however there will be in class exercises the first half (and possibly in subsequent weeks) using a program called Raptor. Raptor requires a PC, so if you have a Mac it is especially important you come to class the weeks we use Raptor if you have a Mac.

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 and active participation are essential if students are to receive maximum benefit from the class. Participation requires preparation including completion of reading, labs, assignments, assessments and exams by the due dates. All due dates will be mentioned on the class blackboard. Students are responsible to submit all work on or before the due dates. It is the students' benefit to use their time wisely whether it is in preparation for class, during scheduled class, or in the lab. When students are in any COD lab environment, they should abide by the college policies. Questions, comments and suggestions are welcome.

Exams and make-up policy: There will be two 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.

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 April 19th.** After that date you will no longer be allowed to drop the course.

Classroom behavior: 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 set all electronic devices such as cell phones or pagers to silent mode. No cell phone talks are allowed in the classroom. If you must take the call, please continue your conversation outside of the classroom and make it short so you will not miss your lectures

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/collage 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 – 3:15 Lecture/discussion
- 3:15 – 3:25 Break
- 3:25 – 4:25 Lecture/discussion /assignment help/Class Review
- 4:25 – 5:00 After class Homework/question session as needed

Grading and Evaluation Criteria

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

NOTE: Due dates for all homework and Quizzes are as assigned in blackboard

Points will be distributed in the following manner:

Programs/Assignments 23 x 10 pts/each – 2 x 15 pts	260
Quizzes 13 x 10 points each	130
Tests (50 points for midterm and 70 points for Final)	120
(Extra Credit) – there will be additional programs for extra credit throughout the semester	Up to 40 points
Total	550

Final Grades will be assigned using the following **point scale** at end of class:

Accumulated Points	Grade	Percentage
450 – 550	A	>= 90 %
400 – 449	B	80 – 89
350 – 399	C	70 – 79
300 – 349	D	60 – 69
239 or lower	F	< 60 %

2/** NOTE an **Incomplete grade** is **not allowed** in this class and the final drop date is April 19th

Tentative Course Outline

Class # and date	Topics	Chapters to read and Prog Exercises at END of chapters - SEE BB FOR ALL HOMEWORK AND QUIZ DUE DATES
1 Feb 18	Course introduction – sample program BB for class. Review syllabus Class student Survey to determine your experience and objectives for course. Chapter 1 Introduction	Class introduction. Syllabus HW 1: Student Survey Class pre test Chap 1 highlights FUZZY HW 2: ASCII Code. Ch 1 Programming Exercises 1-3 (Prog EX 1-3) your name in ASCII Code – see chart in book p 747. Capital 1 st letter and rest lowercase
2 Feb 20	Chapter 2 Input, Process, Output Pseudocode, Flowcharts Python language for a simple Program.	Chapters 2 Example: Will show Prog EX 2-16 in pseudo code and flowchart in Raptor in class HW 3,4: Prog EX 2-6 Raptor in class group exercise, 2-8 in pseudo code.
3 Feb 25	Chapter 2 review and Questions Chapter 3 Modules – Divide and Conquer	Chapter 2, finish up and review Chapter 3(103-116 thru Hierarchy Charts) HW: Prog EX 3-2 in Raptor in class
4 Feb 27	Python Learning a programming language. Chapter 4 Decision Structure and Boolean Logic and Flowcharts	Chapter 4 HW 5: Prog EX 4-5 in class group exercise Raptor and hand drawn flowchart
5 Mar 3	Chapter 4 finish up and review Chapter 5 Repetition Structure (looping). Continue on Repetition Structure. Pseudocode, Flowcharts, Pub Crawl flowchart.	HW 6: Prog EX 4-6 in Python Chapters 5, Do Prog EX 5-5 in class Raptor and Python HW 7: Prog EX 5-9 in class Raptor group exercise plus hand drawn flowchart
6 Mar 5	Chapter 5 finish up and review Continue on Repetition Structure. Pseudocode, Flowcharts	More Python loop examples HW 8: Prog EX 5-3 in Python Pub Crawl – practical flowchart example
7 Mar 10	Chapter 3 117-end of chapter - Local vs Global variables	Chapter 3 rest Will do Prog EX 3-9 IPO diagram and Python in class
8 Mar 12	Finish up Chapter 3 Chapter 6 Functions passing variables Built in functions Turtle package.	HW 9: Prog EX 3-4 Chapter 6 functions turtle. HW 10 Prog EX, 6-7 in class
9 Mar 17	Chapter 6 Functions passing variables -finish	HW 11: 6-12 (Rock-Paper-Scissors) Extra Credit – 5 points your own turtle graphics
10 Mar 19	Chapter 7 Input validation loops –13 pages Practice Midterm – handout	In class design game of Craps with Raptor and program game. Prog EX 7-5 in class – edit Rock, Paper, Scissors
11 Mar 24	Review practice midterm. In class Homework help	HW 12: Prog EX 7-1 in class

12 Mar 26	Midterm chapters 2-7 - 50 points	
Mar 31	SPRING Break – No Class	
Apr 2	SPRING Break – No Class	
13 April 7	Chapter 8 Array/List and Flowcharting. Goes with Repetition – loops and arrays like pizza and beer or pizza and pop.	Chapter 8 Arrays Start HW 13 Prog EX 8-3 in class group homework parallel arrays
14 April 9	Finish up Chapter 8 and review	HW 14: Prog EX 8-2 (write lottery number horizontally) Tic Tac Toe in class – Extra Credit 5 points write winner module Extra Credit 5 points (Lo Shu Magic Square)
15 April 14	Chapter 9 Sorting and Searching Arrays	Chapter 9 notes HW 15: Prog EX 9-2 in class (only use 5 names not 20 as problem states and binary sort function as given in class and NOT Python’s intrinsic sort HW 16: 9-4 use Python Intrinsic sort and binary search given in class
16 April 16	Chapter 10 Files	Examples of Python programs with sequential files HW 17,18: Prog EX 10-1 in class,10-2(just print count of names in the file)
17 April 21	Chapter 11 Menu Driven programming and debugging	Chapter 11 Book example on Metric Conversion HW 19: Prog EX 11-1 (Language Translator) in class
18 Apr 23	Chap 12 Text Processing Chap 14 OOP start	Example Prog EX 12-2 scan for periods. HW 20: Prog EX 12-1 backward string in class – this program algorithm can be done at least 4 different ways – do all 4 for 3 points extra credit
20 Apr 28	Chap 14 OOP	Chapter 14 Object Oriented Programming Pets example Lynda Video on OOP HW 21,22: Prog EX: 14-2 and 14-3 14-2 in class
21 Apr 30	Finish up Chapter 14 OOP and review Practice Final Exam handed out Chapter 15 GUIs start	Finish HW 14-2 in class
22 May 5	Chapter 15 GUIs	Chapter 15 GUI GUI Example of Metric conversion program HW 23,24: Prog EX: 15-3 in class 15-4 on your own
23 May 7	Finish up Chapter 15 GUI and review Posttest Practice exam review	Homework help 5 points EC if 50 or higher
24 May 12	Final Exam	
25 May 14	Help session with homework if needed	All homework/extra credit due 11:59 PM

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.
2. Late assignments are not acceptable unless on medical (with a doctor's letter) or emergency situation (with proof).
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.
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 above. 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.