HARRISBURG UNIVERSITY
OF SCIENCE AND TECHNOLOGY

CISC 504-90
Principles of Programming Languages

October 28, 2017 – February 23, 2018

Instructor: **Muazzam Ali MS**
Email: MUAAli@my.harrisburgu.edu

Office Hours: Online via Moodle Discussion Forum (to be checked as frequently as possible)

Meeting time and location:

Lecture: Thursday 7:00 – 8:00 PM EST (Online Via Adobe Connect)

CATALOG DATA:

This course serves as an introduction to computer programming of scientific applications, using the high-level Python language. The exposition is example- and problem-oriented, where the applications are taken from mathematics, numerical calculus, statistics, physics, biology, and finance. The course teaches procedural programming as well as object-oriented programming. College mathematics is a required background, and it is advantageous to study classical and numerical one-variable calculus in parallel with reading this book. Besides learning how to program computers, student will also learn how to solve mathematical problems, arising in various branches of science and engineering, with the aid of numerical methods and programming. By blending programming, mathematics and scientific applications, the course lays a solid foundation for practicing computational science.

COURSE PREREQUISITE:

- This course is designed for students who have little or no experience in Programming.
- If you are already proficient in programming this may not be a course for you. I would suggest to talk to the program director to learn about other options.

COURSE OBJECTIVES:

- Obtain a firm understanding of the introduction to the principles of Programming using Python.
- A familiarity with the process of writing and debugging a program.
- Learn about the process of transforming a problem definition to its solution through programming.
- A clearer image of what avenues for students in the field of programming and Computer Science.

TEXTBOOK and REFERENCES

The following book is required as the course textbook:

- "A Primer on Scientific programming in Python 2nd Edition", by Timothy J. Barth, Springer-Verlag Berlin Heidelberg, 2011.

HARRISBURG UNIVERSITY
OF SCIENCE AND TECHNOLOGY

COURSE SCHEDULE:

THIS LIST REPRESENTS THE INITIAL PLAN FOR CISC 504. PLEASE NOTE THAT IT IS MERELY A *PLAN*. ACTUAL DATES MAY CHANGE DUE TO WEATHER, ILLNESS OR OTHER UNFORESEEN PROBLEMS. CHANGES WILL BE ANNOUNCED IN CLASS AND POSTED ON THE COURSE'S MOODLE WEB PAGE AS REQUIRED

Week	Date	Topic	Readings	Assignment
1	Thus Nov 02 2017	Course and Tools Introduction & Computing with Formulas	Chapter 1	
2	Thus Nov 09 2017	Computing with Formulas Cont	Chapter 1	Programming Assignment 1 Due Nov 11
3	Thus Nov 16 2017	Loops and Lists	Chapter 2	Programming Assignment 2 Due Nov 18
4	Thus Nov 23 2017	Break		
5	Thus Nov 30 2017	Functions and Branching	Chapter 3	Programming Assignment 3 Due Dec 02
6	Thus Dec 7 2017	Input Data Mid Term Discussion and Questions	Chapter 4	
7	Thus Dec 14 2017	Input Data and Error Handling	Chapter 4	Programming Assignment 4 Due Dec 16
8	Thus Dec 21 2017	Array Computing	Chapter 5	Programming Assignment 5 Due Dec 23
9	Thus Dec 28 2017	Break		
10	Thus Jan 4 2018	Curve Plotting	Chapter 5	Programming Assignment 6 Due Jan 6
11	Thus Jan 11 2018	File String and Dictionaries	Chapter 6	Programming Assignment 7
12	Thus Jan 18 2018	Introduction to Classes	Chapter 7	
13	Thus Jan 25 2018	Introduction to Classes	Chapter 7	Programming Assignment 8
14	Thus Feb 01 2018	Random Numbers and Simple Games	Chapter 8	
15	Thus Feb 08 2018	Random Numbers and Simple Games Object Oriented Introduction	Chapter 8	Programming Assignment 9
16	Thus Feb 15 2018	Object Oriented Programming	Chapter 9	Programming Assignment 10
17	Thus Feb 22 2018	Last Class Final Exam		

	SELF LEARNING
10	Sequences and Difference Equations
11	Introduction to Discrete Calculus
12	Introduction to Differential Equations
13	A Complete Differential Equation Project
14	Programming of Differential Equations

HARRISBURG UNIVERSITY
OF SCIENCE AND TECHNOLOGY

ASSIGNMENT, EXAM, QUIZ DESCRIPTIONS, AND GRADING:

There are programming assignments, Quizzes, midterm exam, and a final exam. Please consult course Moodle page to see due dates for these activities:

Midterm and Final Exams – You will receive pertinent instructions with the specifications in the test book. There will be (1) of each. Each exam will be worth a maximum of **(25) points** of the final grade.

Pre-requisite Assignment Test - You need to conduct all 3 attempts to earn **(10) point** for this activity.

Pop Quizzes – Although these may be plentiful or scarce, they will all only total **(10) points** combined. These will be used if I see a lack of paying attention or I feel that I need to test you in order for you to better understand a concept. Quiz will be open mostly on Sunday, pay attention to Moodle.

Programming Assignments - There will be programming assignments (the instructor will choose from the end of each chapter of the textbook) all worth a **(40) points** total of the final grade. Programing exercises/assignments are due each Saturday except during midterm final and holidays. Rubrics for the projects are added below.

Class Participation - The following guidelines to express the focus of interactive teaching styles:

- Brainstorming - Typically performed in group sessions. The process is useful for generating creative thoughts and ideas. Brainstorming helps students learn to pull together.
- Think, Pair and Share - Establish a problem or a question. Pair the students. Give each pair sufficient time to form a conclusion. Permit each participant to define the conclusion in his or her personal voice. Student are also would explain a concept while the other student evaluates what is being learned. Apply different variations of the process.
- Buzz Session - Participants come together in session groups that focus on a single topic. Within each group, every student contributes thoughts and ideas. Encourage discussion and collaboration among the students within each group. Everyone should learn from one another's input and experiences.
- Incident Process - This teaching style involves a case study format (mostly programming project), but the process is not as rigid as a full case study training session. The focus is on learning how to solve real problems that involve real people. Small groups of participants are provided details from actual incidents and then asked to develop a workable solution.
- Your attendance will be rewarded a maximum of hundred (100) points of the final grade. Classes will start on time and end as scheduled. Please be there time, and try as much as you can not to miss a class otherwise you are going to miss a lot.
- You will attend each class and actively participate in the discussions during class. If you are uncomfortable with public speaking, or if English is not your native language, please talk to the instructor in the first two weeks of the course to establish ways to make you more comfortable in speaking and interacting with other students (your peers).
- You are responsible for all the readings, even if the material is not explicitly covered in class. You should read the class materials prior to class and be prepared to discuss and ask questions

about the readings and assignments. You should also re-read the material after class as not every topic will be covered during class time.

- Take notes as I'm teaching, as I usually let my students you have their notes with them for all exams and quizzes. These notes will help you better understand the issues, concepts and techniques that are being presented and they are the most valuable assets of the course.
- I anticipate that for each lesson you will need to budget about 2 hours. 4 hours are anticipated for each project, and with your test preparation this implies that you need to budget about 120 hours of out-of-class time over the course of the semester. This time estimate is a guide and you may need to budget more. For example, if the material is new to you or difficult to comprehend, it will require more of your time.
- All work must be completed and handed in on Moodle and is to be submitted to the instructor before lesson starting (after the due date). No late work will be accepted. Late means after the class has begun. Note that a computer's failure is not an excuse (it represents poor planning on your part).
- All class credit-related electronic mail must be done using Harrisburg's electronic mail service and the student's assigned Harrisburg University ID. By 'credit-related' I mean all work to be evaluated for credit. Any work submitted through a different mail system will NOT be accepted.
- When individual work is assigned it should be done by you, alone.
- Students who participate in University-sanctioned events (like connection leaders, or athletics) must make prior arrangements and give the instructor ample notice. Missing class for practice is not advised.

COLLABORATION POLICY

The policy is simple: unless otherwise noted in the assignment itself, feel free to collaborate with each other on all the individual coding assignments, but note with whom you collaborated. It is very easy to get a copy of the codes online or from old solutions or course. Please do not copy and paste or borrow solutions without knowing exactly what is going on.

- Our first concern is for what the students in the class learn. We assume that everyone in the class is here because they want to learn, and will behave in a manner consistent with that goal.
- Working with other students on assignments often enhances the learning process. That is why we have a relatively liberal collaboration process. Getting honest evaluations and critiques of your work is also part of the learning process. That is one of the reasons we want you to submit your work individually, and why it is important that the work you submit is truly yours.
- Collaborative learning works best when the students working together have roughly the same level of knowledge and skill. Each participant in the collaboration should contribute more or less equally to solving the problem. When one student is consistently showing another how to do things, it is not a true collaboration. When one student bases their solution on the completed work of another, it is certainly not collaboration.

- We are certainly not opposed to students who understand things helping out those who don't, but that is not the same as collaboration. Also, it is important to understand the difference between helping someone to learn something, and helping them to get a problem set done. The former can be accomplished by explaining how some algorithm works. The latter can be accomplished by providing a ready solution. That is not permitted by our collaboration policy.

STATEMENT ON ACADEMIC INTEGRITY

According to the University's Student Handbook: Academic integrity is the pursuit of scholarly activity free from fraud and deception, and is the educational objective of this institution. Academic dishonesty includes, but is not limited to cheating, plagiarism, fabrication of information or citations, facilitating acts of academic dishonesty by others, unauthorized possession of examinations, submitting work of another person, or work previously used without informing the instructor, or tampering with the academic work of other students. Any violation of academic integrity will be thoroughly investigated, and where warranted, punitive action will be taken. Students should be aware that standards for documentation and intellectual contribution may depend on the course content and method of teaching, and should consult the instructor for guidance in this area.

Honor Code - We as members of Harrisburg University community pledge not to cheat, plagiarize, steal, or lie in matters related to academic work. As a Community of Learners, we honor and uphold the HU Honor Code.

Problems Arise

Problems happen to people when they are least expected. If any problems arise that you expect could impact your work in CISC 504 -- PLEASE CONTACT ME AS SOON AS POSSIBLE! I want to see every student succeed -- but I can only help if I know as soon as possible!