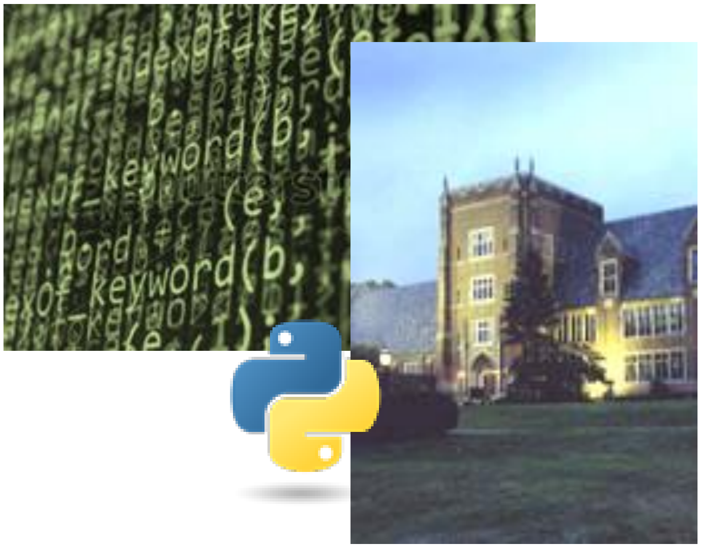
Programming I

MIS 146 Fall Term 2017
MTWF 11:00-11:50 Main Lab



Professor:	Patrick M. Kelly	
Office:	401 Old Main	
Phone:	824-2174	
E-mail:	pkelly@mercyhurst.edu	
Office Hours:	Mondays:	9:00-10:15
	Tuesdays:	12:00-1:00
	Wednesdays:	9:00-10:15
	Thursdays:	11:00-12:00
	Fridays:	9:00-10:30

Meeting times by appointment may be arranged as well.

Blackboard

I do keep *Blackboard* current. Grades, documents from class, reading assignments, etc. will all be posted as they become available. Also, I may want to share files with you for us to work on together during class; for convenience, I will post these files on *Blackboard* and have you access them from there.

I'd also ask you to please check that the e-mail address listed for you on *Blackboard* (by default, your Mercyhurst account) is the account that you use regularly. You might want to change it on *Blackboard* if this is not the case.

Prerequisite:

A minimum score of 46 on the ALEKS placement exam is required for this course.

Text

Programming with Python 3, which is an online interactive textbook. You will need an account on the zyBooks website (zybooks.com) to access this text. Be sure to use your name@lakers.mercyhurst.edu email address for this account. Once you have the account created, enter the code

MERCYHURSTMATH146KellyFall2017

to subscribe for the specific book for our class.

Software

The programming language of choice for this course is Python. The current version is Python 3.6.2. We have Python installed on each of the computers here in the lab (and surrounding area), but you're also encouraged to put Python on your own computer. This is free software (go to python.org to download).

Course Description:

This course is designed to introduce the student to the discipline of computer programming. Specifically, the computer programming language of Python will be utilized in this course; as such the student will develop and hone programming skills

using Python. More importantly, though, the emphasis will be on the development of good programming habits and transferable programming skills.
Topics will include fundamental programming concepts such as:

- loops,
- decisions,
- lists,
- functions,
- file input/output,
- arrays,
- objects
- classes, and
- inheritance.

Student Learning Outcomes:

The course provides an introduction to programming, basic algorithmic techniques and implementation of computer programs.
Specifically, by the end of the semester the student should be able to:

1. understand computer architecture and data representations (variables, representation of numbers and character strings) ;
2. learn basic algorithmic problem-solving techniques (decision structures, loops, functions) ;
3. communicate problem-solving algorithms, in English as well as in pseudocode;
4. write short programs in Python using conditional statements, loops, input, variables, and list manipulations;
5. debug small programs and sue exception handling in Python to avoid runtime errors; and
6. write and use functions to perform smaller tasks in the context of a larger program.

Grading:

Your final course grade will be compiled from several factors: reading assignments, lab activities, and four exams (three midterms and one final). The contribution of each factor toward your course grade will be as follows:

Reading assignments	5%
Lab activities	30%
Exams #1-3	15% each
Final Exam	20%

Your final percentage, then, will determine your letter grade, according to the grading scale for the Mathematics and Information Technologies Department:

A	90-100%	B+	87-89%
B	80-86%	C+	77-79%
C	70-76%	D	60-69%
F	0-59%		

Reading Assignments:

It is important for you to complete the assigned readings for each lesson. These lessons will primarily be “flipped,” where you are to read about the upcoming concepts **before** the corresponding class session, and then in class we will explore those ideas together and solidify your understanding. A standing **deadline** for these reading assignments will be **10:45** on the day of the lesson (15 minutes before class). What specific sections need to be read before which class sessions will be laid out between the schedule here on the syllabus and updated announcements/postings on Blackboard.

As you read each section, there will be “Participation Activities” to complete. These Participation Activities will be the way in which you demonstrate your completion (and understanding) of the reading assignments. The Participation Activities are where you will earn the points that go toward the 5% of your final grade.
***Note**: If you score 75% or higher on any individual Participation Activity, your score for that grade will get bumped up to 100%.*

You will also notice “Challenge Activities” in several of the sections. These are not for points; you may complete them at your discretion.

Lab Activities:

Tuesdays will be “lab days.” During these class sessions, you will be given directions for a task you are to complete during that class. Each of these programming projects will be due that day. These lab assignments will be opportunities for you to demonstrate a hands-on understanding of the concepts we cover in lecture.

Your lowest three grades for these lab activities will be dropped at the end of the semester.

Exams:

Each of the tests will be traditional “paper & pencil” exams. Thus, the focus will be much less about actually performing any kind of programming on the computer and more about the concepts, ideas, and theory of programming. You can expect the questions asked on exams to be more conceptual and less procedural (more T/F, multiple-choice, and short-answer essay questions than any hands-on task to be completed).

A Few Policies:

- You are responsible for all that is covered (including announcements) in class even if you are absent.
- If you must miss an exam or in-class activity, a make-up is only considered when (a) notice is given (when possible) prior to the missed assessment, (b) there is a valid reason for missing, and (c) **the make-up assessment is completed prior to the next class session**.
- Office hours are not for reteaching lessons or catching you up on something you missed.
- You are responsible for all the material in a given section unless told otherwise. Use the course schedule and listed reading assignments as a guide.
- A prerequisite for additional help outside the classroom is regular class attendance.

The Computer Lab:

Are you mature enough to be in a class that's held in a computer lab? Please resist the temptation to hit up YouTube, Netflix, Reddit, or other such distractions that sitting at a computer can present.

Realize that much of what I have to show you in this course are things that I have to *show you*—typically on the overhead screen. If you're not paying attention while I'm demonstrating something, do not expect me to explain it to you once you're done doing your own computer work.

Cell Phones:

Before each class session begins, please try to remember to turn your phones off so as not to cause a disruption during class (and do realize that even the sound of a vibrating phone can be disruptive). Furthermore, cell phones are not permitted at all during exam situations.

Services:

In keeping with college policy, any student with a disability who needs academic accommodations must call Learning Differences at 824-3017 or stop by Old Main room 314, to arrange a confidential appointment with the Disability Services Director during the first week of classes.

(Tentative) Course Schedule:

Read Ahead	Day	Concepts
	Wednesday, 8/23	**Read After**: §1.1 – 1.3
§1.4 – 1.6	Friday, 8/25	Chapter 1: Introduction to Computing
§2.1 – 2.2	Monday, 8/28	Introduction to programming
§2.3 – 2.8	Tuesday, 8/29	Lab #0; Installing Python
§2.9, §2.11– 2.14	Wednesday, 8/30	Errors, Objects, variables, assignments
§3.1 – 3.2	Friday, 9/1	Variable types
No class!	Monday, 9/4	*Labor Day*
§3.3 – 3.4	Tuesday, 9/5	Lab #1
§4.1 – 4.3	Wednesday, 9/6	Functions
§4.4 – 4.7	Friday, 9/8	Further work with functions
§5.1 – 5.5	Monday, 9/11	Chapter 5: Modules
	Tuesday, 9/12	Lab #2
§6.1	Wednesday, 9/13	If-else statements
§6.2 – 6.4	Friday, 9/15	Multiple if-else and other conditionals
	Monday, 9/18	Catch-up/Review
	Tuesday, 9/19	Lab #3
Exam #1	**Wednesday, 9/20**	*Chapters 1-6*
§7.1 – 7.2	Friday, 9/22	Strings
§7.3 – 7.5	Monday, 9/25	Formatting strings, string methods
	Tuesday, 9/26	Lab #4
§7.7 – 7.8	Wednesday, 9/27	Splitting and joining strings, representing

§8.1 – 8.2	Friday, 9/29	While loops
§8.3 – 8.4	Monday, 10/2	More with ‘while’, counting
	Tuesday, 10/3	Lab #5
§9.1	Wednesday, 10/4	Lists
§9.2 – 9.3	Friday, 10/6	Dictionary basics, data types
§9.4 – 9.7	Monday, 10/9	Further work with lists
	Tuesday, 10/10	Lab #6
§9.8	Wednesday, 10/11	Command-line arguments
No class!	Friday, 10/13	*Mid-Semester Break*
§10.1 – 10.2	Monday, 10/16	For loops
	Tuesday, 10/17	Lab #7
§10.3 – 10.5	Wednesday, 10/18	Nesting loops, ‘while’ vs. ‘for’
§10.6 – 10.9	Friday, 10/20	Further loop concepts
§10.11, 10.13 – 10.14	Mon., 10/23	Iterating over a list, loops modifying lists
	Tuesday, 10/24	Lab #8
	Wednesday, 10/25	Catch-up/Review
Exam #2	**Friday, 10/27**	*Chapters 7-10*
§11.1 – 11.2	Monday, 10/30	Reading and writing files
No class!	Tuesday, 10/31	*Advising Day*
	Wednesday, 11/1	Lab #9
§11.4 – 11.6	Friday, 11/3	Command-line work with files, ‘with’, CSV
§12.1 – 12.4	Monday, 11/6	Functions as objects, loops in functions, scope
	Tuesday, 11/7	Lab #8
§12.5 – 12.6	Wednesday, 11/8	Function arguments, keywords, parameters
§12.7 – 12.8	Friday, 11/10	Argument lists, multiple function output
§13.1 – 13.3	Monday, 11/13	Exception handlers, raising exceptions
	Tuesday, 11/14	Lab #11
§13.4 – 13.6	Wednesday, 11/15	Exceptions with functions, custom exceptions
§14.1	Friday, 11/17	Classes
§14.2 – 14.3	Monday, 11/20	Classes to group data, class methods
§14.4 – 14.5	Tuesday, 11/21	Class examples
No classes!	Wednesday, 11/22	*Thanksgiving Break*
	Friday, 11/24	
§14.6 – 14.7	Monday, 11/27	Operator overloading, class constructors
	Tuesday, 11/28	Lab #12
§14.8 – 14.9	Wednesday, 11/29	Class customization, overriding methods
	Friday, 12/1	
Exam #3	**Monday, 12/4**	*Chapters 11-14*
	Tuesday, 12/5	Fun with Python
	Wednesday, 12/6	Fun with Python
	Friday, 12/8	Review
Final Exam	**Fri., 12/15 at 10:30**	*Everything!*