CSCI 150 - Foundations of Computer Science

Hendrix College

Lec 01: MWF 8:10am - 9:00am (A1)
MCReynolds 110

Lab L1: R 8:10am - 11:00am (L4)
Bailey Library Snoddy Lab

Instructor: Dr. Mark Goadrich

Contact Info
goadrich@hendrix.edu
MCReynolds 313

MW 10:30 - 11:45, T 9:30 - 11, F 1-2pm
or by appointment

Lec 02: MWF 11:10am - 12:00pm (A4)
MCReynolds 315

Lab L2: W 1:10pm - 4:00pm (L8)
Bailey Library Snoddy Lab

Instructor: Dr. Gabriel Ferrer

Contact Info
ferrer@hendrix.edu
MCReynolds 312

Office hours:
By appointment: `drferrer.youcanbook.me`

Overview | Syllabus | Labs | Projects | Exams | Grading

Course Details

Textbook

Think Python
by Allen Downey, 2012
Olin College, MA
This textbook is open-source; we have reorganized and edited it to match our course syllabus.

Software

Python 3.6 PyCharm Edu

This class is BYOL. Bring a laptop computer to class with you every day, unless there is a test. We have a small number of computers available for students who are unable to do so. We will do programming exercises every class period.

Overview

Introduction to solving computational problems, including the fundamentals of computer programming. Topics include imperative programming constructs (variables, loops, conditionals, functions, recursion), basic object-oriented constructs (classes, objects), and some fundamental algorithms and data structures (dictionaries, arrays, linked lists, basic sorting). Student learn these concepts through studying the Python programming language.

At the end of the course, you will be expected to be able to:

- Read, understand and execute a computer program written in Python.
- Read a set of requirements for a computer program in English, and write a short Python program (100 lines or less) that corresponds to them.
- Test a Python program and identify and fix programming errors.
- Identify some errors in a Python program without testing it.
- Without using a computer, write a very short Python code fragment (10 lines or less) that correctly implements a set of requirements.
- Understand and apply variables, loops, strings, lists, conditionals, and functions.
- Write programs to perform mathematical calculations.
- Understand the concept of a module.
- Write a Python program that is separated into at least two modules.
- Understand the concepts of class and object, and distinguish between them.
- Write a Python program including objects of at least one student-designed class.
- Write and understand appropriate comments in a Python program.
- Understand the concept of an algorithm and compare the efficiency of different algorithms for a simple task.

Disabilities

It is the policy of Hendrix College to accommodate students with disabilities, pursuant to federal and state law. Students should contact Julie Brown in the Office of Academic Success (505.2954; brownj@hendrix.edu) to begin the accommodation process. Any student seeking accommodation in relation to a recognized disability should inform the instructor at the beginning of the course.

Academic Honor

Please refer to the CSCI Academic Integrity Policy.

Revisions

After assignments are returned, you are welcome to revise and resubmit your work. Each submitted revision will be graded anew, the original and revised grades will be averaged to produce a new grade for that assignment. Revisions may be submitted anytime until the start of the final exam period.

Extensions

No late work will be accepted. Any work not submitted on time is a zero. However, you may submit a solution after the deadline to qualify under the revision policy. In effect, this means that late work can earn up to half credit.

Syllabus

We will be covering most of the material in the textbook, approximately one new chapter each week. You should view your textbook as another perspective on the material presented in class and covered in the labs. We will also be using additional supplemental material such as relevant web-pages and background material for the lab assignments. Readings will be assigned before material will be covered in class. You are expected to review the material and come to class prepared.

- Chapter 1 - August 25
- Chapter 2 - August 28
- Chapter 3 - August 28
- Chapter 4 - September 1
- Chapter 5 - September 18
- Chapter 6 - September 20
- Chapter 7 - September 25
- Chapter 8 - September 25
- Chapter 9 - September 29
- Chapter 10 - October 4
- Chapter 11 - October 16
- Chapter 12 - October 20
- Chapter 14 - October 25
- Chapter 15 - November 1
- Chapter 16 - November 3
- Chapter 17 - November 8
- Chapter 18 - November 10

Quizzes and Participation and Homework

Throughout the semester, there will be short quizzes and homework exercises covering material from the previous class. Each quiz and homework exercise will be awarded a certain number of points. A total of 300 points will be available over the course of the semester. These quizzes, participation, and homework will comprise 15% of your final grade. Earning 150 points suffices for full credit.

Item	Name	Due	Points
HW 0	Who are you?	Aug 25	5
HW 1	Communication and Origami	Aug 28	12
QUIZ 1	Quiz 1	Aug 28	8
HW 2	Boolean Logic Puzzles	Sep 6	10
QUIZ 2	Quiz 2	Sep 6	5
QUIZ 3	Quiz 3	Sep 11	10
PRACTICE EXAM 1	Exam 1 Practice	Sep 13	15
HW 3	DNA Strings	Sep 29	12
HW 4	While Exercises	Sep 29	10

HW 5	For Exercises	Oct 6	18
PRACTICE EXAM 2	Exam 2 Practice	Oct 9	15
HW 6	Functional Practice	Nov 3	15
HW 7	Zen Readings	Oct 27	0
QUIZ 4	Quiz 4	Nov 3	10
PRACTICE EXAM 3	Exam 3 Practice	Nov 17	15

Labs

Much of your experience with programming in this course will be through weekly labs, which will comprise 25% of your final grade. Each lab will be assigned in lab with time allotted to work through the materials, and will be due at the start of the next lab. All labs are weighted equally within the lab portion of your final grade.

Lab attendance is required. Labs take place in the **Snoddy Computer Lab**, in the Bailey Library.

You will be handing in your lab work via Moodle. Instructions to do so will be included in each lab.

On these labs, you may work with a partner on the lab assignments if you choose. Their name must be listed on any code you hand in as joint work.

Lab	Name
1	Minecraft
2	Kepler and Newton
3	Diagnosing Heart Disease
4	Today in History
5	Guess My Number
6	Mutation is the Word
7	Caesar's Secrets
8	Fractal Recursion
9	Movie Reviews
10	Die Hard III
11	Processing
12	Stuckness
13	Civic Data Mashup

Projects

You will have three projects in this course, one about every five weeks, for a total of 35% of your final grade. These projects will cover concepts we have discussed in class and in labs, and will be due approximately one week after they are assigned.

You must work individually on these projects. You may discuss concepts and ideas with your classmates, but the code you turn in must be your own. You will be graded not only on correctness, but also technique, documentation and evaluation of your solution. Further details on the grading standards and handin instructions for each project will be given when they are assigned.

Project	Name	Assigned	Due
1	Civic Assistance Application 5%	Sep 17	Sep 25
2	Word Games 10%	Oct 11	Oct 25
3	Your Choice 20%	Nov 6	Finals Day (Dec 11)

Exams

There will be three in-class exams, the first worth 5% and the last two each worth 10% of your final grade. They will consist of short answer along with writing and debugging code.

- Exam 1: Sep 15, covering input/output, math, numerical data and conditionals
- Exam 2: Oct 11, covering functions, while and for loops, lists and strings
- Exam 3: Nov 20, covering recursion, dictionaries and object-oriented programming

Grading

Your final grade for this course will be based on the Labs, Projects, Quizzes, Exams and Participation described above.

Grading Scale	
90-100	A
80-89	B
70-79	C
60-69	D
0-59	F

Weights	
Labs	25%
Projects	35%
Quizzes, Participation, Homework	15%
Exam 1	5%
Exam 2	10%
Exam 3	10%