COMP 1900 CS 1 – Section 001
Introduction to Programming – Fall 2025
Bill Baggett, PhD

Instructor Contact Information:

- wbaggett@memphis.edu - Please send emails from your University email address to wbaggett@memphis.edu with the subject "COMP 1900." I will try to respond within 24 hours.
- Office: Dunn Hall 390

  Office Hours
  Please email me to schedule an appointment.

Lecture Meeting Times/Locations:

Section	Meeting Times	Location
001	Mon/Wed 12:40pm-2:05pm	Dunn Hall Room 351

Lab Meeting Times/Locations:

Section	Meeting Times	Location	Lab TA
101	Wed 1020-1220	DH 232	Yanliang Qi, yqi2@memphis.edu
102	Thu 1440-1640	DH 232	Nacim Mohamed Osman, nmosman@memphis.edu
103	Wed 1420-1620	DH 232	Hasan Mashrique, hmshrque@memphis.edu
104	Thu 1220-1420	DH 232	Hasan Mashrique, hmshrque@memphis.edu
105	Fri 1020-1220	DH 232	Tyler Andrew Howell, thowell3@memphis.edu
106	Fri 1220-1440	DH 232	Tyler Andrew Howell, thowell3@memphis.edu

Catalog Description:

COMP 1900 – CS 1: Introduction to Programming (4) Overview of computer science as a field; problem-solving strategies with emphasis in fundamental programming skills, primitive data types, control structures, arrays, strings, I/O, basic recursion, documentation, testing and debugging techniques; introduction to object-oriented concepts. Three lecture hours, two laboratory hours per week. PREREQUISITE or COREQUISITE: MATH 1910 or MATH 1421 (or MATH 1830 for COMP minors)

Note on Prerequisites:

Although COMP 1900 does not assume you have any prior programming experience, it moves quite briskly. If programming is brand new for you, you'll likely find it more difficult to keep up. We recommend COMP 1800 (Problem Solving with Computers) as a warm-up course for COMP 1900 if you are just starting out with programming.

Course Outcomes:

After completing this course, students should be able to:

1. Read and write code involving variables and assignments.
2. Read and write code involving conditionals.
3. Read and write code involving loops.
4. Read and write code to manipulate strings, lists, and dictionaries.
5. Read and write code to manipulate one- and two-dimensional lists.
6. Write programs to define functions and make function calls with various input and output types.
7. Be able to decompose code into sections using functions.
8. Be able to define simple classes.
9. Be able to create objects and invoke method calls.

Course Website:

Course materials and grades will be posted to the Canvas system at https://memphis.instructure.com/

Required Text:

Online text through zyBooks. A subscription costs $77 and will last until Jan 14, 2026.

1. Click any zyBooks assignment link in Canvas. (Do not go to the zyBooks website and create a new account)
2. Subscribe

If you're retaking the course or have otherwise subscribed to the zyBook previously, you may be eligible for a free or reduced-cost subscription. Please email support@zybooks.com to take advantage of this.

Additional Free Online Resources:

- Official Python tutorial: https://docs.python.org/3/tutorial/
- Educative: https://www.educative.io/courses/learn-python-3-from-scratch
- W3Schools: https://www.w3schools.com/python/
- CodingBat (interactive code practice): https://codingbat.com/python

Evaluation:

Lecture Attendance	100 pts
zyBook Reading Assignments	100 pts
zyBook Coding Assignments	100 pts
Lab Assignments	310 pts
Midterm Exam	160 pts
Final Exam (Comprehensive)	300 pts

Your final grade is determined by (your total points on all graded items) / 1000. Note that the highest possible percentage grade is 107% since the points add up to 1070. This gives you some built-in buffer in case your second cousin's ex-wife's brother-in-law has an event that forces you to miss an assignment, or a temporary zombie apocalypse happens to just your neighborhood, or whatever. ***This also means I'll be strict about enforcing assignment deadlines. Please don't beg for late credit.***

Note that you receive a single combined grade for COMP 1900; there's *no* separate grade for the lab section unlike some science courses.

Grading Scale: Letter grades will be determined from your total points as follows:

A+: 96% and above; **A**: 90-95%
B+: 87-89%; **B**: 81-86%; **B-**: 79-80%
C+: 77-78%; **C**: 71-76%; **C-**: 69-70%
D+: 67-68%; **D**: 60-66%
F: Below 60%

Assignments:

This is a programming course, and the only way to get better at programming is to do a lot of it. There will be many assignments throughout the semester to give you hands-on practice. Altogether, assignments make up *about half your grade* for this course. You cannot pass unless you do them!

The assignments fall into different categories:

1. **zyBook reading assignments** are short exercises selected from the required textbook. These are done on your own, to further reinforce what was covered during lecture. You should complete all **Participation** and **Challenge** activities from the assigned sections. These sections will be graded at the end of the semester, but it is highly recommended to complete each week's sections.
2. **zyBook coding assignments** are selected programming exercises from the required textbook. Some of these will be assigned for you to work on during class. Any assigned exercises that you do not complete during class should be done on your own.
3. **Lab assignments** allow you to explore a topic more deeply and can be more involved than the classwork and zyBook assignments. Each lab assignment has two components: a short in-lab that must be completed before leaving lab for the day, and a longer lab homework that you'll have at least one week to complete on your own.

More About the Labs:

- There is no new material covered during lab sessions. The labs are meant for you to get hands-on practice with programming each week. Attendance at lab sessions is REQUIRED.
- Lab sessions are supervised by TAs. Their contact info is posted on your lab section on Canvas.
- Each lab has two components: a short in-lab assignment and a longer lab homework. Your lab TA will distribute the in-lab assignment during each lab meeting. This assignment must be completed before you leave lab for the day. If you finish the in-lab early, you are welcome to start working on the lab homework. You have at least one full week to complete each lab homework.
- Lab TAs are there to help you during lab sessions. Please don't hesitate to ask them for assistance, on the in-lab assignment as well as lab homework. The TAs may also be available outside of lab sessions via email, and/or office hours.
- You get a single combined grade for the COMP 1900 lecture and lab. Lab assignments give you much-needed practice with programming and make up a substantial portion of your final grade, so be sure to turn them in!
- There are NO LAB MEETINGS during the following weeks:
  - Aug. 25-29 (first week of class)
  - Oct. 13-17 (fall break week)
  - Nov. 17-28 (the week before and during thanksgiving)
  - Dec. 1-5 (last week of the class)

Time Expectations:

Programming definitely has a learning curve, and many people find this to be a demanding class. Most students should expect to spend 2-3 hours per week outside of class per hour of credit. Since this is a 4-hour class, that translates to *8-12 hours per week of work outside of class*. If you are unable or unwilling to devote this time, I strongly recommend that you postpone taking the course until you can. There is no substitute for hands-on experience to improve your programming skills!

Attendance:

It is crucial that you attend class (both lecture and lab) regularly, especially if this is your first experience with computer programming. ***I will directly take attendance each lecture***.

Late/Makeup Policy:

All assignments are expected to be completed and turned in on schedule. Due dates will be clearly indicated for each assignment. Late assignments are NOT accepted except in extreme circumstances. Likewise, makeup exams will be given only under extreme circumstances. *If you feel that your circumstances warrant a late work submission or a makeup exam, get in touch with me as soon as possible. Be prepared to show some kind of documented proof of your situation.*

Plagiarism/Cheating Policy:

By the end of this course, you are expected to be a competent programmer. This is important for success in future courses, and more importantly so that you can get a job later! To improve your programming skills, you must get plenty of practice yourself. As such, *all grade items (unless specifically indicated otherwise) must be individual efforts*. Although you are welcome to work in study groups, NEVER submit any code that you did not write yourself.

Examples of ACCEPTABLE behavior:

- Discussing the general solution approach to an assignment with other students, then writing your code to solve the problem individually
- Using Internet resources to help with the concepts behind an assignment, then writing your own code that incorporates what you've learned

Examples of UNACCEPTABLE behavior:

- Submitting the same code as another student. Making trivial changes like changing variable names and/or order of functions does not hide this.
- Copying code that you did not write yourself (including code from "homework help" sites like Chegg, and generative AI tools like ChatGPT)
- Copying someone else's code during an exam

I have a zero-tolerance policy against cheating. Plagiarized code is very obvious. If I catch you submitting code that you did not write yourself, the first offense will result in a 0 for that grade item and referral to the Office of Student Accountability. Further offenses will possibly result in a failing grade in the entire course. *Please don't put me (or yourself) in this situation.*

Getting Help:

Although I expect your work for this class to be done individually, I encourage you to seek help if you get stuck:

- Contact your lab TA. They are there to help you.
- Contact me! I'm very willing to provide hints without giving away the solution. I can be reached via email.
- Online tutoring: The UofM offers free online tutoring through the Educational Support Program (ESP): https://www.memphis.edu/esp/onlinetutoring.php

Miscellaneous Policies:

Email - Please check your University of Memphis email account at least once a day, as that is my primary means of communicating with you outside of class.

Student Disabilities - If you have a disability that may require assistance or accommodations, or if you have any questions related to any accommodation for testing, note taking, reading, etc., please contact me as soon as possible. You must contact the Disability Resources for Students office (901.678.2880, drs@memphis.edu, https://www.memphis.edu/drs/) to officially request such accommodations / services.

COMP 1900 Course Schedule (Section 001, Mon/Wed 12:40pm-2:05pm, Dunn Hall Room 351):

Date	Lecture Material	In-Labs	Lab Homework
8/25 8/27	Intro to computers, CS, number systems, first Python programs		Lab 0 HW: Lab contract
9/01 9/03	*NO CLASS – Labor Day* Variables and expressions	In-Lab 1 - Basic Python Programs	Lab 1 HW: Number systems, basic programs
9/08 9/10	Variables and expressions Conditionals	In-Lab 2 - Variables and Expressions	Lab 2 HW: Variables and expressions
9/15 9/17	Conditionals	In-Lab 3 - Conditionals	
9/22 9/24	Loops	In-Lab 3b - More Conditionals	Lab 3 HW: Conditionals
9/29 10/01	Loops	In-Lab 4 - Loops	
10/06 10/08	Review for midterm Midterm	In-Lab 5 - More Loops	Lab 4 HW: Loops
10/13 10/15	*NO CLASS – Fall Break* Functions		
10/20 10/22	Functions	In-Lab 5b - Nested Loops	Lab 5 HW: More loops
10/27 10/29	Strings	In-Lab 6 - Functions	Lab 6 HW: Functions
11/03 11/05	Lists and tuples	In-Lab 7 - Strings	Lab 7 HW: Strings
11/10 11/12	Dictionaries and sets	In-Lab 8 - Lists	Lab 8 HW: Lists and dictionaries
11/17 11/19	Intro to object-oriented programming concepts		Lab 9 HW: OOP
11/24 11/26	Intro to object-oriented programming concepts *NO CLASS - Thanksgiving*		
12/01 12/03	Review for final		

MIDTERM EXAM:
Wednesday, Oct 8th, Topics: Everything up to and including loops

FINAL EXAM:
Wednesday, Dec 10th, 10:00am - noon, Topics: Everything
See the full final exam schedule here: https://www.memphis.edu/registrar/calendars/