

Syllabus

Programming I – COMP/CENG 150
3:00 MWF, Science 207
Spring 2017

Instructor: Frank McCown
Contact: 501-279-4826, HU Box 10764, fmccown@harding.edu
Home Page: <http://www.harding.edu/fmccown/classes/comp150-s17/> (Syllabus, useful links)
Office Hours: Science 208: 10 – 11 am and 4 – 5 pm daily, or by appointment

Course Description

Fundamental concepts of problem solving and computational algorithms. Using the C++ programming language, a study will be made of language syntax, program control flow, algorithm implementation, modular program design, and arrays. No prior programming experience is required. COMP 151 continues where this class ends. No textbook is required although a CodeLab subscription is.

Student Learning Outcomes

The student will be able to...

1. Formulate algorithmic solutions to problems in a structured flowchart.
2. Identify and eliminate syntax and logic errors in a program.
3. Use functions to build modular programs.
4. Design and develop programs that make use of single-dimensional arrays.
5. Test and verify that a program satisfies specific requirements.

Attendance

Those who attend class regularly will usually do much better than those who miss frequently, so you are expected to be in class *every day*. You may have two “free skips,” but each unexcused absence after that will result in 1% being subtracted from your final grade. Attendance will be taken daily at the beginning of class. If you are late, it is your responsibility to see me after class. Absences that are excused (illness, school sponsored trips, etc.) will not be held against you. After missing a class, it is **your responsibility** to determine what you missed and what homework might be due the following day.

Exams

Three hour long exams will be given along with a cumulative final exam. If you are unable to take an exam as scheduled due to a serious illness or some other emergency, it is **your responsibility** to email or call me and leave a message **before** the exam or as soon as you are physically able. If an official school function takes you out of class on an exam date, it is your responsibility to make arrangements *one week prior* to the exam as to when you will take the exam. Usually it will be given early, not late. Makeup exams for excused absences will be given, but a penalty of up to 75% will apply for unexcused absences, at the teacher’s discretion. Phones must be turned off and put away.

Homework and Labs

There will be numerous homework assignments and in-class labs. Homework and labs are usually due at the beginning of the class period following the day they are assigned. The homework is to be completed *individually*. We will be using CodeLabs for some of the homework assignments. This is an online system which requires you to pay a fee of \$25. More information on CodeLabs will be given later in the semester.

Most labs are to be completed in *pairs* (2 people). Pair programming has been shown to have a number of benefits including increased personal satisfaction and fewer errors¹, and it helps most students when first learning to program. When working in pairs, both students must work together on a *single* computer, and both must write *approximately half* of the code. **No code can be written without their partner present and watching.** Both of you should understand completely what is being written. If it is not possible for a pair to find time to work together outside of class, each person can finish the lab independently.

¹ See *All I Really Need to Know about Pair Programming I Learned in Kindergarten* (2000) for more information on effectively using pair programming,

Programming Projects

Three large programming projects will be assigned, and you will have one to two weeks to complete each project. Programming projects are major assignments which require dedicated effort and time to complete. You will use **Visual Studio 2015** to write the programs; it is installed on all machines in the classroom and 201 lab. To obtain a free copy of VS to install on your own computer, click the link on the class website to download *Visual Studio Community*.

You may work independently on your projects or in pairs (with your assigned lab partner or someone else). Just like the labs, both people must work together on a *single* computer, and both must write *approximately half* of the code. **No code can be written without their partner present and watching.** Both people should understand completely what is being written. When you submit a program that has been written in pairs, you must document at the top of the program the names of both individuals who worked on the program. Only one student should submit the program.

Extra Credit

A maximum of 2% extra credit can be earned and applied to your final grade.

1. **Computing Seminar:** You will receive **0.1%** points extra credit for each Computer Seminar that you attend. Seminar meets every Friday at 7:00 am in Science 113. The first seminar will begin the 4th week of the semester. There will be approximately 11 seminars, thus allowing you to increase your final grade by 1.1%. See the schedule at <http://www.harding.edu/comp>
2. **Giving Blood:** Donating at the Red Cross blood drives will earn you **0.2%** added to your final grade each time you donate. Donate as many times as you'd like, and give me a signed note confirming your donation each time you donate.
3. **The McChallenge:** **1%** will be added to your final grade for the completion of a program which will be made available to you later in the semester. The program will be due the Friday before final exams. You can skip the program and still get the 1% added to your final grade if you beat me in a game of basketball, tennis, racquetball, Halo, chess, Trivia Pursuit, or any other sport/game that I know how to play. If you lose, you still may complete the program to get your 1%. Only one challenge per semester, and all challenges must be made *before* the final week of class. Come by my office to schedule a time to play.

Grades

Final grades will be computed as follows:

Exams	30%
Programming Projects	30%
Homework and Labs	20%
Final Exam	20%

Standard letter grades: A = 90-100%, B = 80-89% C = 70-79%, D = 60-69%, F = 0-59%

Late work: A maximum of 10% will be taken off *each day* (not each class period) a program or assignment is late, up to 50%. Every day is counted, including weekends. Nothing late more than 1 week will be accepted.

Final grades are not rounded unless the student has given significant effort which is evidenced by regular attendance, completion of nearly all homework assignments, working well with partners in completing labs, etc.

Miscellaneous

1. Notes are available to you on Easel. If you want to print them out, please do so *before class, not during class* so you don't disrupt others.
2. Tutoring is available in the Sci 201 Computer Lab on most evenings (check the schedule which is posted in the lab). Tutors are junior or senior computer science majors who have completed this course and more advanced programming courses.
3. To be successful in this course, the typical student will need to spend at least **two hours outside of class** for every hour in class studying, completing homework, working on projects, and preparing for exams. If you do not have six hours every week outside of class to dedicate to the course, you should take it some other semester when you can make that commitment.
4. You must check Canvas and Easel regularly for announcements, class discussions, and assignments. Canvas is where you can ask questions and give help to others on homework, labs, and projects and where class recordings can be accessed.
5. Everyone is expected to hold the **highest standard** of personal conduct and **integrity**. Cheating in all its forms is inconsistent with Christian faith and practice and will result in sanctions up to and including dismissal from the class with a failing grade. Homework should be completed *individually* (not in teams or pairs), and it should be *your* work, not the work of someone else. Projects should also be completed individually although they may be completed using pair programming. One thing that you should *never*

do is give someone your source code to look at as this often leads to cheating. Come by during office hours (or we'll arrange a time) for assistance on programs. Also take advantage of the tutor who will be available several times a week.

6. Please adhere to the **dress code** as spelled out in the Student Handbook. This includes men removing caps while in class. Please wear shoes to class (flip flops are OK).
7. There is **no food or drink** allowed in the lab. However, I will allow you to bring in a drink with a lid until we have a spill.
8. Lab computers may be used during class to **take notes and write programs** when specifically allowed by the instructor. Until the instructor gives you permission, your monitor is to remain **off**. Students who use lab computers for other purposes will not be allowed to use the lab computers.
9. Silence your phones, and **put them away**. It is very distracting to me and those around you when you text in class.

Computer science is one of the most fascinating fields you can study and currently has the **most stable and satisfying job markets**. It is, however, a science, and as such requires a dedicated effort to master. It cannot be mastered without **persistence** and **practice**. You should expect to struggle with some of the difficult concepts in this course, but do not give up. Those who do the best in this course attend class regularly, turn in homework and assignments on time (because they don't procrastinate), and seek help from the tutor or myself when in a rut. Remember that I am here to help you.

Assessment

Harding University, since its charter in 1924, has been strongly committed to providing the best resources and environment for the teaching-learning process. The board, administration, faculty, and staff are wholeheartedly committed to full compliance with all criteria of the Higher Learning Commission of the North Central Association of Colleges and Schools. The university values continuous, rigorous assessment at every level for its potential to improve student learning and achievement and for its centrality in fulfilling the stated mission of Harding. Thus, a comprehensive assessment program has been developed that includes both the Academic units and the Administrative and Educational Support (AES) units. Specifically, all academic units will be assessed in reference to the following Expanded Statement of Institutional Purpose: **The University provides programs that enable students to acquire essential knowledge, skills, and dispositions in their academic disciplines for successful careers, advanced studies, and servant leadership.**

Students with Disabilities

It is the policy for Harding University to accommodate students with disabilities, pursuant to federal and state law. Therefore, any student with a *documented disability* condition (e.g. physical, learning, psychological, vision, hearing, etc.) who needs to arrange reasonable accommodations, must contact the instructor and the Disabilities Office at the *beginning* of each semester. (If the diagnosis of the disability occurs during the academic year, the student must self-identify with the Disabilities Director *as soon as possible* in order to get academic accommodations in place for the remainder of the semester.) The Disabilities Office is located in Room 205 in the Student Center, telephone: (501) 279-4019.

Schedule

The following schedule is subject to change but gives you an idea of how the class will progress:

Week 1 Jan 9	Introductions History of computing Hardware and software basics	Week 7	Misc C++ features Project 1 Nested ifs & loops	Week 11	Lab 5 - Debugger Pass by referene Lab 6 - Functions
Week 2	No class on MLK Day Algorithms and flowcharts	Week 8 Feb 27	Chars & complex conditions Lab 3 – Nested ifs and chars Exam 2	Week 12 Apr 3	Project 3 Exam 3
Week 3	Flowcharting	<i>Spring Break</i>		Week 13	Intro to arrays Arrays and loops
Week 4 Jan 30	C++ syntax intro Lab 1 - Compile and run Exam 1	Week 9	Data validation Project 2 for loops and switch statements	Week 14	Arrays and functions Lab 7 - Arrays
Week 5	Arithmetic expressions and ops Converting flowcharts into C++	Week 10	Lab 4 – Ave, max, min Intro to functions Functions that return values	Week 15	Sorting algorithms Lab 8 – Sort
Week 6	if and if-else statements while and do-while statements Lab 2 – Flow chart to code			Week 16 May 1	Final Exam

