

CS 210 – Fundamentals of Programming I

Spring 2018 – Syllabus

Instructor

Dr. Don Roberts
KC-259, 488-2667
Email: roberts@refactory.com or roberts@evansville.edu
Web: <http://csserver.evansville.edu/~droberts>

Class Hours

01 - TTh 8:00 AM-9:50 AM, KC-267
02 - TTh 1:00 PM-2:50 PM, KC-267

Office Hours

MW, 10:00AM –11:00 AM, 1:00 PM–3:00 PM

Catalog Data

Emphasizes problem-solving techniques used in the analysis and design of software solutions, including structured top-down design, abstraction, good programming style, debugging and testing. Programming constructs covered include control structures, functions, and basic and aggregate data types. Introduction to recursion and dynamic allocation.

Credit Hour Policy

This course meets the federal requirements of 15 in-class hours plus an expected 30 hours of out-of-class work per credit hour (DEF 1).

Objectives

Learning problem solving techniques used in programming software solutions including structured design, good programming style, testing strategies, and debugging strategies. Note that you will be learning and using C as a vehicle towards these goals, but this course is not about learning C in its entirety. More specifically:

- Students will be able to write programs using selection, repetition, functions, and dynamic allocation in one major high level language.
- Students will be able to write programs using the aggregate data types arrays, strings, and structs.
- Students will design programs using the fundamentals of functional decomposition.
- Students will be able to apply testing and debugging strategies.

- Students will be able to construct software solutions that use structured analysis and design with good programming style.
- Students will be proficient using an IDE, such as Xcode or Code Blocks, to implement and debug programs.

Prerequisites

None

Required Textbook

Jeri R. Hanly, Elliot B. Koffman, *Problem Solving and Program Design in C, 7ed*, Addison-Wesley, 2013, ISBN 978-0-13-293649-1.

Daily and Weekly Requirements

Assigned daily videos with embedded quizzes and reading. In-class programming assignments. Additionally, there will be an outstanding 1- or 2-week project that you are expected to work on solo.

Class Format

This is going to be a *flipped* class. What that means is that there will be an assigned video or videos that students are expected to watch before coming to class. To measure this, the videos will have embedded quiz questions to answer. You will not get credit for watching the videos after class has started. There will also be assigned readings from the textbook to complete before class. In class, the first hour will typically consist of problems, usually from the book, to work on with the help of other around you and the instructor. This is the time to ask questions and get help with concept that you are having difficulty with. Once these are completed, there will also be an individual project due each week that you are expected to work on during class time. Typically, you will not be able to complete it entirely within class, so you will most likely have to spend some outside time finishing it. You are expected to complete the projects on your own.

Programming Projects

Generally, these will be given out on Tuesday, and will be due at midnight one week later.

Exams and Evaluations

There will be two in-class written exams and a comprehensive final written exam. In addition, there will be a lab programming practical exam. The purpose of the practical exam is to demonstrate mastery in using the C programming language. Final grades will be based on the following distribution:

20 %	Comprehensive Final Exam
20 %	2 in-class exams
10 %	Lab programming practical exam
10 %	Video Quizzes
10 %	In-class Homeworks
30 %	Programming Projects

Attendance, Missed Exams, Late Homework and Machine Problems

Video quizzes must be completed by the beginning of class on the date specified. In class homeworks must be completed before the class period, but ideally should be completed in the same period. Machine problem are due by 11:59pm of the date specified. Any machine problems handed in after lecture has started are considered late. The following late penalties will be applied:

One day late	10%
Two days late	10% + 20% = 30%
Three days late	10% + 20% + 30% = 60%
Four days late	Do the math...

Video quizzes and in-class homeworks cannot be taken late as these are typically graded pass/fail.

Valid excuses for missing exams and handing assignments in late include illness, family emergencies, religious observances, official UE events, etc. They do not include work conflicts, studying for other classes, staying home an extra day over the weekend, or {dog, virus, space aliens, zombies}{ate, wiped out, abducted}{my homework, my computer, me, my brain} (in any combination).

The instructor will rely on your integrity on getting work excused. If you have a valid excuse, put it in writing, sign your name to it, and give it to the instructor. For religious observances and UE events, you must inform the instructor that you will be absent **before** the absence occurs, otherwise it will be considered an unexcused absence.

Excused work must be made up within two class meetings. Late work will not be allowed. Exceptions will be made for serious or prolonged illness, or other serious problems. Please note that it is your responsibility to take care of missed or late work.

Honor Code

All students at the University of Evansville agree to the University honor code: "I will neither give nor receive unauthorized aid, nor will I tolerate an environment that condones the use of unauthorized aid."

- In-class homework exercises are for you to gain experience and practice. You may collaborate with your classmates, but each student should submit a solution in his/her own words that reflect his/her understanding of the solution. You will eventually be required to demonstrate your knowledge of the material on the exams, so it is better that you attempt the problems on your own.
- Solutions shall not be copied from Internet or other sources. Discovery will result in a 0 for that assignment for all parties involved for the first offense. A second offense will result in failure of the course. A third offense will result in dismissal from the program. Please note, that this penalty is cumulative throughout your career at UE.
- For all machine problems, you are expected to code the solutions yourself. You may not observe or copy another person's code or solution.
- Any attempt to compromise the automatic grading system will result in immediate failure of the course.

Disability Policy

It is the policy and practice of the University of Evansville to make reasonable accommodations for students with properly documented disabilities. Students should contact the Office of Counseling and Health Education at 488-2663 to seek services or accommodations for disabilities. Written notification to faculty from the Office of Counseling and Health Education is required for academic accommodations.

Schedule

Week	Date	Tuesday	Thursday
1	[2018-01-08 Mon]	Chapter 1 Introduction Compilers	Chapter 2 Overview of C
2	[2018-01-15 Mon]	Chapter 3 Functions	Chapter 3 More Functions
3	[2018-01-22 Mon]	Chapter 3 Even More Functions	Chapter 4 Selection
4	[2018-01-29 Mon]	Chapter 5 Loops	Chapter 5 Cont. More Loops
5	[2018-02-05 Mon]	Chapter 6 Program Design	Debugging Exam 1 Review
6	[2018-02-12 Mon]	Exam 1	Chapter 11 Files
7	[2018-02-19 Mon]	Chapter 7 Arrays	Practical Exam
8	[2018-02-26 Mon]	Chapter 7 Cont'd Arrays	Chapter 7 cont. Multidimensional Arrays
9	[2018-03-05 Mon]	SPRING BREAK	SPRING BREAK
10	[2018-03-12 Mon]	Chapter 8 Strings	Chapter 8 cont. More Strings
11	[2018-03-19 Mon]	Chapter 10 Structs	Chapter 10 More structs
12	[2018-03-26 Mon]	Exam 2 Review	Exam 2
13	[2018-04-02 Mon]	Chapter 12 Separate Compilation	Chapter 9 Recursion
14	[2018-04-09 Mon]	Chapter 13 Dynamic Arrays	Chapter 13 Dynamic Allocation
15	[2018-04-16 Mon]	Chapter 13 Data Structures	Chapter 14 Processes / Threads
16	[2018-04-23 Mon]	Review	