

# CS 210 – Fundamentals of Programming I

Spring 2017 – Syllabus

## Instructor

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

## Class Hours

Section 1 – MW 3:00–4:50PM, KC-267  
Section 2 – TTh 1:00–2:50PM, KC-267

## Final Exam Time

Section 1 – Fri. 4/28 – 2:00PM–4:00PM  
Section 2 – Tues. 5/2 – 11:00AM–1:00PM

## Office Hours

MTWTh, 8:30AM–11:00AM

## 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.

## Objectives

Learning how to survive the impending zombie apocalypse by applying 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, it's about survival in the face of innumerable hordes of undead. 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 understand the fundamentals of functional decomposition and design.

- Students will demonstrate an understanding of 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 Visual Studio, 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, 8ed*, Addison-Wesley, 2016, ISBN 978-0-13-401489-0.

## **Recommended Textbook**

Brooks, Max, *The Zombie Survival Guide: Complete Protection from the Living Dead*, Three Rivers Press, 2003, ISBN 1-400-049628.

While there will not be any assignments given from this book, it is the definitive reference for preparing for the imminent zombie uprising.

## **Daily and Weekly Requirements**

Assigned daily reading and weekly homework or programming assignments. During each lab time, there will be an outstanding project that you are expected to work on solo.

## **Class Format**

This class meets for 2 hours in a lecture/lab format. Typically, the first hour of the class will consist of lecture and example programs demonstrated by the instructor. The second hour of the class is typically for you to work on the currently assigned machine problem. You are expected to use this time for this purpose. The only valid reason for leaving early is if you have already completed the assignment. During this time the instructor is available to answer your questions regarding the program and its design. This is incorporated into the design of this course and if you do not take advantage of this, several key concepts will be missed (and your grade will probably end up reflecting this).

Additionally, you are expected to read the assigned chapter(s) from the textbook. There is not enough time during lecture to cover all the material in the depth necessary for mastery. You are responsible for all of the material in assigned chapters even if it is not explicitly covered in lecture.

## **Programming Projects**

Generally, these will be given out the first time the class meets in a given week, and will be due before class one week later. These projects are expected to be completed on your own. See the section regarding the Academic Honor Code below for more information.

## 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 and the programming environment used in this course. Therefore, it is necessary to score a minimum of 60% on the practical exam to pass the course. Students who fail to do so and are otherwise passing, will be given a second opportunity to pass the practical exam at the end of the term with a 10% penalty. Final grades will be based on the following distribution:

20 %	Comprehensive Final Exam
20 %	2 in-class exams
10 %	Lab programming practical exam
10 %	Homework
40 %	Programming Projects

## Attendance, Missed Exams, Late Homework and Machine Problems

Homeworks are due at the beginning of class on the date specified. Machine problem are due by 11:59 a.m. of the date specified. Any assignments handed in after the due date and time 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...

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 are expected to adhere to the University's Honor Code regarding giving and receiving authorized aid.

- 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.
- Projects are to be your own work. 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.

- You may not observe another person's code or solution. You are expected to code the programming projects yourself. You may discuss generic approaches that you took to the problem, but you may never share actual code.
- Any attempt to compromise the automatic grading system will result in immediate failure of the course.

## Schedule

Week	Date	Monday	Tuesday	Wednesday	Thursday
1	<2017-01-09 Mon>	Chapter 1 Introduction Compilers	Chapter 1 Introduction Compilers	Chapter 2 Overview of C	Chapter 2 Overview of C
2	<2017-01-16 Mon>	NO CLASS MLK Day	Chapter 3 Functions	Chapter 3 Functions	Chapter 3 More Functions
3	<2017-01-23 Mon>	Chapter 3 More Functions	Chapter 4 Selection	Chapter 4 Selection	Chapter 5 Loops
4	<2017-01-30 Mon>	Chapter 5 Loops	Chapter 5 Cont. More Loops	Chapter 5 Cont. More Loops	Chapter 6 Program Design
5	<2017-02-06 Mon>	Chapter 6 Program Design	Chapter 11 Files	Chapter 11 Files	Chapter 7 Arrays
6	<2017-02-13 Mon>	Chapter 7 Arrays	Chapter 7 Multidimensional Arrays	Chapter 7 Multidimensional Arrays	Exam 1 Review
7	<2017-02-20 Mon>	Exam 1 Review	Exam 1	Exam 1	NO CLASS
8	<2017-02-27 Mon>	Chapter 8 Strings	Chapter 8 Strings	Practical Exam	Practical Exam
9	<2017-03-06 Mon>	NO CLASS Spring Break	NO CLASS Spring Break	NO CLASS Spring Break	NO CLASS Spring Break
10	<2017-03-13 Mon>	Chapter 8 More Strings	Chapter 8 More Strings	Chapter 10 Structs	Chapter 10 Structs
11	<2017-03-20 Mon>	Chapter 10 More Structs	Chapter 10 More Structs	Chapter 11 More Structs	Chapter 11 More Structs
12	<2017-03-27 Mon>	Chapter 12 Separate Compilation	Chapter 12 Separate Compilation	Chapter 10 Recursion	Chapter 10 Recursion
13	<2017-04-03 Mon>	Exam 2 Review	Exam 2 Review	Exam 2	Exam 2
14	<2017-04-10 Mon>	Chapter 14 Dynamic Allocation	Chapter 14 Dynamic Allocation	Chapter 14 Dynamic Arrays	Chapter 14 Dynamic Arrays
15	<2017-04-17 Mon>	Chapter 14 Data Structures	Chapter 14 Data Structures	Chapter 15 Processes / Threads	Chapter 15 Processes / Threads
16	<2017-04-24 Mon>	Review	Review	READING DAY	