

CYBR210: Computer Programming with a High Level Language

College of Graduate and Continuing Studies, Norwich University

[Print This Page](#)

Course Description

3 credits

This course provides an introduction to computing and programming with C++. Topics covered include program design, compilation, variables, data types, control structures, functions, pointers, memory, classes, abstraction, and object-oriented concepts. Debugging, quality software engineering techniques, and security concerns will be discussed throughout.

As software becomes more ubiquitous in every aspect of our lives, it is imperative that cyber security professionals have a fundamental understanding of common programming paradigms. In this course, we will use the C++ programming language to study structured programming and introduce students to program design. We will explore the entire life cycle of programs, from design conception to program execution on a computer. This will be a challenging course as there is a lot of material to cover and getting your hands dirty in programming for the first time can be quite intimidating. But fear not, because with determination and commitment I am sure you can do this! Be sure to read the text, engage in discussions, ask questions, and practice! Programming takes practice. Welcome to the course and I wish everyone a successful and enlightening seminar.

Pre-requisites: None

Course Outcomes

The overall goal of this three credit seminar is to understand the basics of high level computer programming in C++. At the conclusion of this seminar, students will have:

1. An understanding of the software development lifecycle (SDLC)
2. The ability to solve basic programming problems and implement simple algorithms in C++
3. Knowledge of the logic, syntax, and control structures of structured programming and object oriented programming in C++
4. A basic understanding of programming languages, compilers, linkers, executable files, and the role of the operating system in program execution.

Students who complete the course will be able to:

1. Discuss the difference between compiled programming languages and interpreted programming languages
2. Design, write, compile, debug, and execute basic C++ programs
3. Discuss the compilation process
4. Describe and implement various C++ data types
5. Describe and implement common C++ control structures and library functions
6. Design and implement functions
7. Describe computer memory, variable scope, and pointers
8. Design classes and user-defined data types
9. Implement basic input/output techniques in C++
10. Implement arrays and vectors
11. Discuss abstraction, encapsulation, data hiding, and modular decomposition

12. Describe object-oriented programming
13. Algorithmic design and problem solving
14. Basic software design and documentation
15. Implement exception handling.

[Instructor Contact Policy](#)

I will answer all e-mails and discussion questions as promptly as possible, typically within 24 hours. Whenever possible, please post questions to the weekly discussion forums, as other students may benefit from our discussion. This also limits me answering duplicate questions individually over e-mail.

[Weekly Outline](#)

Week	Topic	Readings	Assignments
Week 01	Programming and Problem Solving in C++	Chapters 1, 2, and 3	Week 1 discussion; Problem Set 1
Week 02	I/O, Conditions, Logic, and Selection Control	Chapters 4 and 5	Week 2 discussion; Problem Set 2
Week 03	Looping	Chapters 6 and 7	Week 3 discussion; Problem Set 3
Week 04	Functions	Chapters 8 and 9	Week 4 discussion; Problem Set 4
Week 05	User Defined Data Types, Structs, Pointers, and Arrays	Chapters 10 and 11	Week 5 discussion; Problem Set 5
Week 06	Classes, Abstraction, and Objects	Chapter 12	Week 6 discussion; Problem Set 6
Week 07	Dynamic Data, Inheritance, and Polymorphism	Chapters 14 and 15	Week 7 discussion; Problem Set 7
Week 08	Templates, Overloading, Exception Handling, and Vectors	Chapters 16 and 17	Week 8 discussion; ; Problem Set 8; Final Project

[Required Textbooks](#)

The required textbook(s) are:

- Dale, N., & Weems, C. (2014). *Programming and problem solving with C++, comprehensive edition* (6th ed.). Jones & Bartlett Learning.

In lieu of a writing handbook, we strongly recommend that students use the [Purdue Online Writing Lab](#) as a primary writing resource. Your instructor will provide additional writing guidance.

[Discussion Questions](#)

There will be a discussion question each week that requires a response from each student. In order to receive full credit for the participation, responses should be complete and reflect understanding of the course material. There is no minimum word count. As long as the response completely addresses the question at hand, then the student

will receive full credit. Discussion are designed to promote collaborative learning. Students are encouraged to respond to each other's posts and create questions of their own. The discussion forum is also our place to discuss the week's material and assignments.

Assignments

Lectures and Readings

The text book contains a lot of information and I do not expect you to memorize everything in the text. So do not get caught up in trying to master every technical detail in the book during your weekly readings. Reading through the chapters and lecture slides will provide exposure to the material. Your weekly problem sets will help you focus your efforts on the important concepts.

Problem Sets

Each week will include a required problem set to be submitted by every student. Problems will address the week's learning objectives. Submit your assignment as a PDF document. Complete each problem completely. Problems requiring programming should include source code, screen shots of the working program, and a brief explanation for full credit. If you cannot resolve some of your program's errors, then include your error messages and discuss your theories as to why the program is not running correctly. You can still receive close to full credit without a fully working program! Use complete sentences, proper spelling, and proper grammar. If you are obtaining your information from the text or course materials, there is no need to site your information. That being said, do not copy verbatim from the text or lecture, but instead use your own words as best as possible. Avoid using resources outside of the classroom, such as the Internet, to solve the problems.

Final Project

The Final will cover material from throughout the course. It includes a major programming piece. I encourage student to begin working on the final as early as week 6 to afford yourself enough time to complete the required tasks.

Late Work

It is important that writing assignments and discussion posts be completed on time. Extensions of deadlines will be given only for serious extenuating circumstances. In the absence of such extensions, assignments may be downgraded for lateness at the discretion of the instructor. Students are encouraged to coordinate with the instructor to request extensions in advance if needed. Extensions will be granted on a case by case basis and are only granted as a result of extraneous circumstances.

Grades

Grades will be based the quality and accuracy of problem set solutions, performance on the final, and contributions to the online discussions. Partial credit is given whenever possible. The course grade is distributed across the components listed below.

As a student in this course, you have a number of responsibilities that will affect the level of learning you achieve. These responsibilities include:

1. Working actively to create a challenging and useful learning experience for yourself, your discussion group and the class as a whole.
2. Encouraging and supporting the learning of each member of the class.
3. Preparing and participating fully in discussions as well as group and/or class activities. and

4. Completing all assigned work on time or making prior arrangements if an absence of late submission is unavailable.

The course grade will be determined as follows:

All activities are graded on **100** points per activity and the overall value for the different groups of activities is worth the percentage shown below.

Activity	# Required	% of Grade
Weekly Discussions	8	15%
Weekly Problem Sets	8	60%
Final Project	1	25%

Letter grades for the course will be based on the following grading scale:

Letter Grade	Percentage	Grade Point
A	93-100%	4.0
A -	90-92.9%	3.7
B +	87-89.9%	3.3
B	83-86.9%	3.0
B -	80-82.9%	2.7
C +	77-79.9%	2.3
C	75-76.9%	2.0
C -	73-74.9%	1.7
D +	70-72.9%	1.3
D	67-69.9%	1.0
D-	63-66.9%	0.7
F	0-62.9%	0.0

For complete information on the Grading Policy for Bachelor Degree students, please refer to the [CGCS Online Catalog](#) (Sub-Section of Catalog on "Grades.")

[Academic Honesty and the Norwich University Honor Code](#)

A student must submit work that represents the student's own original analysis and writing. Copying another's work is not appropriate. If the student relies on the research or writing of others, the student must cite those sources. Words or ideas that require citations include, but are not limited to all hardcopy or electronic publications, whether copyrighted or not, and all verbal or visual communication when the content of such communication clearly originates from an identifiable source. While students are encouraged to seek editing feedback, extensive revisions of one's work by another person is considered a lack of academic honesty, as it is representing another student's work as one's own.

For more information see:

[Academic Dishonesty](#)
[Academic Integrity](#)
[Norwich University Honor Code](#)

Copyright Notice

The content of this seminar contains material used in compliance with the U.S. Copyright Law, including the TEACH Act and principles of "fair use." Materials may not be downloaded, saved, revised, copied, printed or distributed without permission other than as specified to complete seminar assignments. Use of these materials is limited to class members for the duration of the seminar only.

Section 504 of the Rehabilitation Act of 1973/ADA

Please consult [Appendix H: University Policy - Section 504 of the Rehabilitation Act of 1973/Americans with Disabilities Act \(ADA\)](#) for instructions on obtaining an accommodation.

Disclaimer: Please note the specifics of this Course Syllabus are subject to change. Students are responsible for abiding by any such changes. Your instructor will notify you of any changes.

Copyright © Norwich University 2016