

C/C++ Programming

SYLLABUS AND SCHEDULE | Winter 2022

Course description

Programming and problem solving using the C++ programming language. Dynamic memory, classes, linked lists, stacks, queues, and testing. After 30 years, C and C++ remain the most widely used programming languages in the industry, but why?? This course will not only provide the basics of how to program in these languages, it will dive under the hood to show how the constructs are implemented and executed on a computer system.

Course information

Course code: ECEGR 2020
credits: 5
Location: BANN 201
Class times: MW 2:05–3:30pm
Lab times: W 3:40–5:45pm
Classes/Labs: Jan 4 – Mar 14
Final Exam: Mar 17 2:00-3:50pm

Instructor information

Instructor: Ken Rabold
Phone: 206-399-7811
Email: raboldke@seattleu.edu

Text:

The C Book, by Mike Banahan, Declan Brady and Mark Doran, 1991, Addison Wesley.
<https://github.com/wardvanwanrooij/thecbook/releases/latest>

A Complete Guide to Programming in C++, by Ulla Kirch-Prinz and Peter Prinz, 2002, Jones and Bartlett Publishers. ISBN: 0-7637-1817-3

References:

C Programming Language 2nd Edition, Kernighan and Ritchie, 1988, Pearson.
ISBN: 0131103628

The C++ Programming Language, 4th Edition, Bjarne Stroustrup, 2013, Addison-Wesley Professional. ISBN: 0321563840

Tools:

CodeLite: <https://codelite.org/>

MinGW: <https://sourceforge.net/projects/mingw-w64/> Click Files -> MinGW-W64-install.exe

Git tools: <https://git-scm.com/downloads>

Learning outcomes

On successful completion of this course, you will be able to

1. Design and implement computer programs using classes across multiple files, applying good style and appropriate documentation.
2. Explain and appropriately use dynamic memory.
3. Implement algorithms to manipulate linked lists, stacks and queues and use them appropriately in solving problems.
4. Create unit test cases using pre-and post-conditions and boundary values.
5. Use systematic debugging techniques effectively to trace code and identify problems.

Grading procedures and policies

Each lab must be demonstrated to the instructor and written up as a report. Each lab will have a specific due date. The report must contain the following elements.

1. A cover sheet with the name and number of the lab and your name.
2. A document that shows the structure of your program. It describes in plain language the sequence your program goes through to solve the design problem. This may be in the form of pseudocode or flowcharts.
3. A description of how you tested your program. This is not a description of the steps carried out to enter your code and compile or assemble it. It is a description of what you did to convince yourself that your program is so well built that it will always work correctly and anyone could safely run it. Consider how you tested boundary cases in your program.
4. A link to the Git location of your source
5. The lab report is submitted to Canvas
6. All source code is submitted to your personal Git repository

There will be one exam and a final. The final is comprehensive.

Labs: 70%

Exam #1: 15%

Final Exam: 15%

GRADING SCALE

A	100–93	Superior	B–	82–80		D+	69–67	
A–	92–90		C+	79–77		D	66–63	Poor
B+	89–87		C	76–73	Adequate	D–	62–60	
B	86–83	Good	C–	72–70		F	59 or less	Failing

Schedule

Jan 5	01 – Class Overview, Tools Lab #0 – Setup of Tools
Jan 10 Jan 12	02 – Program Structure and Style, Variables, Functions 03 – Arithmetic, Logical, Bitwise Ops, Assignment and Conditionals, Control Reference – C Programming: Chap 9, 10, 12, 13, 15, 16 Lab #1 – Due Jan 26
Jan 17 Jan 19	No class. Martin Luther King Jr. Weekend 04 – Compilation, Preprocessor, Libraries, Input/Output, Working with Files Reference – C Programming: Chap 8, 17, 18, 19, 20 Lab #2 – Due Feb 2
Jan 24 Jan 26	05 – Memory, Arrays, Pointers, Strings, Memory Management 06 – Structures, Unions, Enumerations, Type Casting Reference – C Programming: Chap 23, 24, 25, 26, 27 Lab #3 – Due Feb 16
Jan 31 Feb 2	07 – Single and Double Linked Lists, Stack, Queue 08 – Structure of computer application in memory, CPU execution Lab #4 – Due Feb 23
Feb 7 Feb 9	09 – C in the Real World; Exam Review Exam #1
Feb 14 Feb 16	10 – Why C++, Object Oriented Programming 11 – C++ Classes, Properties, Methods, Constructors, Destructors, this Reference – Programming in C++: Chap 1, 13, 14 Lab #5 – Due Mar 2
Feb 21 Feb 23	No class. Presidents' Day Weekend 12 – C++ String class, Overloading I/O classes, Namespaces Reference – Programming in C++: Chap 4, 9, 11, 18, 19 Lab #6 – Due Mar 9
Feb 28 Mar 2	13 – C++ Polymorphism, Inheritance 14 – C++ Memory Management, Copy Constructor, Virtual Destructor Reference – Programming in C++: Chap 20, 21, 22, 23, 24, 25 Lab #7 – Due Mar 16
Mar 7 Mar 9	15 – C++ Data Structures 16 – Design Patterns, How Stuff Works... Reference – Programming in C++: Chap 28
Mar 14 Mar 17	17 – C++ in the Real World; Final Exam Review Final Exam

Academic resources

My goal is to create a learning environment in which you can be incredibly successful. I will work hard to create and improve the learning environment throughout the quarter based on my own observations of the course and your feedback on what would help you learn more. In return, I ask and encourage you to make the most of this learning opportunity. Please take advantage of the academic support services available to you at the university. Even if you have had excellent study skills in the past, it is easy to slip into suboptimal habits and these services can help you excel in your studies.

LIBRARY AND LEARNING COMMONS

<http://www.seattleu.edu/learningcommons/>

WRITING CENTER

The Writing Center employs undergraduate writing consultants who assist students at all stages of the writing process. Consultants will help students begin writing tasks, organize and develop first drafts, and revise and edit later drafts.

LEARNING ASSISTANCE PROGRAMS

Learning Assistance Programs provide peer tutoring, facilitated study groups, and learning strategy development through scheduled workshops and individual meetings with a learning specialist.

RESEARCH SERVICES

Need help finding research? Save time by starting with your Research Services Librarians. We are eager to help you at any stage of the research process. Contact us if you need help brainstorming keywords, using our databases, finding articles and books or sorting through the information you find on the internet. Students can receive help in person, by chat, phone, or email, or by scheduling a research consultation.

MATH LAB

The Math Lab is a drop-in service available to students enrolled in lower division mathematics courses. Students can stop by the lab to work with a tutor who will assist them with their particular mathematics assignments.

ACADEMIC INTEGRITY TUTORIAL

<https://www.seattleu.edu/academicintegrity/>

General course and university policies

SUPPORT FOR STUDENTS WITH DISABILITIES

If you have, or think you may have, a disability (including an “invisible disability” such as a learning disability, a chronic health problem, or a mental health condition) that interferes with your performance as a student in this class, you are encouraged to arrange support services and/or accommodations through Disabilities Services staff located in Loyola 100, (206) 296-5740. Disability-based adjustments to course expectations can be arranged only through this process.

ACADEMIC POLICIES ON THE REGISTRAR WEBSITE

<https://www.seattleu.edu/registrar/academics/performance/>

Be sure that you understand the following university academic policies, posted on the Registrar’s website:

ACADEMIC INTEGRITY POLICY

ACADEMIC GRADING GRIEVANCE POLICY

PROFESSIONAL CONDUCT POLICY

(only for those professional programs to which it applies)