

Syllabus

CIS 17A C++ Programming

General Course Information

Contact Information

Instructor: [Dr. Mark E. Lehr](#)

Email: mark.lehr@rcc.edu

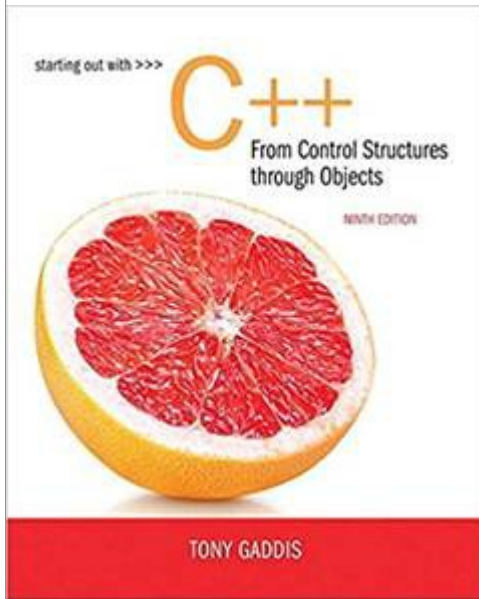
Phone: (951) 222-8260

[Course Description](#)

Course Sequence of Study (Course content related to the Gaddis textbook sections/chapters)

1. Review of C++ concepts. Chapters 1-8
2. Introduction and overview of object-oriented programming and C++. Chapter 13 and 14.
3. Functions Chapter 6 and Chapter 8
4. Pointers Chapters 7, 9 and 10
5. Structures Chapter 11
6. Classes and members Chapter 13
7. Overloading Chapter 6 and 13
8. Constructors and destructors Chapter 14
9. Friends Chapter 14
10. Polymorphism Chapter 15
11. Inheritance Chapter 15
12. Advanced input and output Chapter 12
13. Templates Chapter 16
14. Linked Lists Chapter 17, Stacks and Queues Chapter 18

Textbook: (Yes, if you have an earlier version such as the 8th edition, you will be able to use it in my class, however, if you have an opportunity to obtain the new 8th edition then please do.)



3

Starting Out with C++ : From Control Structures through Objects (Formerly "Standard Version") (8th Edition) (Paperback)

- **Loose Leaf:** 1344 pages
- **Publisher:** Pearson; 9 edition (February 27, 2017)
- **Language:** English
- **ISBN-10:** 0134443829
- **ISBN-13:** 978-0134443829

Student Learning Objectives:

1. Analyze and understand the Object-Oriented C++ environment.
2. Demonstrate the use of specialized terminology, directives, and features of the C++ language.
3. Apply theoretical business, scientific, and mathematical concepts in writing and executing programs in the C++ language using Object-Oriented programming methodology.
4. Demonstrate specified problem-solving skills using the C++ language.

Bottom Line - At the end of the course you will need to demonstrate the ability to create 2 dimensional dynamic arrays of objects that utilize abstract, inherited, and generic programming concepts.

4

5	<p>Course Requirements (assignments as a percentage of grade)</p> <p>Homework -- 20% Assigned at the end of every chapter. Graded as pass/fail for each problem.</p> <p>Projects -- 40% Two projects are required including a written report, full documentation, as well as working code. The first project is worth 20% and the final project is worth 20%.</p> <p>Midterm -- 20% Appropriate problems will be assigned which tests the students knowledge of the subject matter. Requires commented working code.</p> <p>Final -- 20% Will test the students ability to solve problems utilizing all the constructs covered in class. Also, requires commented working code.</p> <p>Note: All assignments, projects, and exams are to be done individually by the student. The homework maybe discussed but each student is responsible for turning in their own assignments.</p>														
6	<p><u>Grading Rubric</u></p>														
7	<p>Grade scale for entire course</p> <table data-bbox="354 926 570 1262"> <tr> <td>90 - 100</td> <td>A</td> </tr> <tr> <td>80 - 89</td> <td>B</td> </tr> <tr> <td>70 - 79</td> <td>C</td> </tr> <tr> <td>60 - 69</td> <td>D</td> </tr> <tr> <td>0 - 59</td> <td>F</td> </tr> </table>	90 - 100	A	80 - 89	B	70 - 79	C	60 - 69	D	0 - 59	F				
90 - 100	A														
80 - 89	B														
70 - 79	C														
60 - 69	D														
0 - 59	F														
8	<p>Course Homework Assignments</p> <table data-bbox="272 1461 1122 1944"> <thead> <tr> <th>Lessons</th> <th>Homework Problems</th> </tr> </thead> <tbody> <tr> <td>1 – Chapters 1-4</td> <td>3.12, 3.13, 4.10 and 2 other problems</td> </tr> <tr> <td>2 – Chapters 4-8</td> <td>5.11, 6.7, 7.6, 8.7 and 1 other problem</td> </tr> <tr> <td>3 – Chapter 9</td> <td>9.2, 9.6, 9.7 and 7 other problems</td> </tr> <tr> <td>4 – Chapter 10</td> <td>10.4, 10.6 and 3 other problems</td> </tr> <tr> <td>5 – Chapter 11</td> <td>11.9 and 9 other problems</td> </tr> <tr> <td>6 – Chapter 13.1-9</td> <td>13.1,13.4 and 3 other problems</td> </tr> </tbody> </table>	Lessons	Homework Problems	1 – Chapters 1-4	3.12, 3.13, 4.10 and 2 other problems	2 – Chapters 4-8	5.11, 6.7, 7.6, 8.7 and 1 other problem	3 – Chapter 9	9.2, 9.6, 9.7 and 7 other problems	4 – Chapter 10	10.4, 10.6 and 3 other problems	5 – Chapter 11	11.9 and 9 other problems	6 – Chapter 13.1-9	13.1,13.4 and 3 other problems
Lessons	Homework Problems														
1 – Chapters 1-4	3.12, 3.13, 4.10 and 2 other problems														
2 – Chapters 4-8	5.11, 6.7, 7.6, 8.7 and 1 other problem														
3 – Chapter 9	9.2, 9.6, 9.7 and 7 other problems														
4 – Chapter 10	10.4, 10.6 and 3 other problems														
5 – Chapter 11	11.9 and 9 other problems														
6 – Chapter 13.1-9	13.1,13.4 and 3 other problems														

	<p>7 – Chapter 13.10-18 13.5,13.6 and 3 other problems</p> <p>8 – Chapter 12 12.7,12.8, 12.11 and 2 other problems</p> <p>9 – Chapter 14.1-4 14.1,14.2 and 3 other problems</p> <p>10 – Chapter 14.5-8 14.3,14.4 and 3 other problems</p> <p>11 – Chapter 15 15.1, 15.4, 15.6 and 7 other problems</p> <p>12 – Chapter 16 16.1, 16.2, 16.3, 16.4, 16.5 and 5</p> <p>Additional homework that might utilize lists, recursions and trees</p>
9	<p>Course Structure – Expectations</p> <p>No late homework assignments, projects, or exams! If an extension is granted it will apply to the whole class.</p> <p>No makeup projects, tests, or finals!</p> <p>Consideration will be given to students that are going out of town on business for the week, etc... Must make prior arrangements with the instructor. Assignment due dates are posted in Canvas.</p> <p><u>Lab Hours</u></p> <p>Lab assignments will be provided for the on-line and night courses. Day courses will need to complete 18 hours of lab in mlk-219.</p>
10	<p>Statement on Accommodations</p> <p>If you have a physical, psychiatric/emotional, medical, or learning disability that may impact your ability to carry out assigned course work, I would urge that you contact the staff in the Office of Disabled Student Services. The Office is located on the Riverside Campus, Room 21 of the Administration Building, or call (951)222-8060 (City Campus), (951) 372-7070 (Norco Campus), (951) 485-6138 (Moreno Valley Campus). DSP&S will review your concerns and determine, with you, what accommodations are necessary and appropriate. All information and documentation is confidential.</p>
11	<p>NET-iquette</p> <p>Like being in a face-to-face class, students in an on-line class must be able to discuss and debate divergent views without ridicule or personal attack. An important part of learning is considering the broad range of views possible on any one subject. You will be engaging in on-line discussions on topics that may yield a diverse array of opinion. Any behavior that is considered offensive in a classroom setting will be considered offensive on-line. This includes, but is not limited to the use of profanity, racial, sexual, or religious epithets, harassing or disrespecting another person on-line. Remember, all discussions are monitored daily.</p>

DIVERSITY STATEMENT

12

Riverside City College School of Business embraces a notion of an intellectual community enriched by diversity with multiple dimensions, including race, ethnicity and national origin, gender, gender identity, sexuality, class, and religion. We are particularly committed to populations that have historically been excluded from equitable participation in the classroom, higher education institutions, and our communities. Individually, we are devoted to addressing our unconscious bias to pave the way for a more inclusive curriculum and learning environment.

Course Requirements

Follow the 16-week sequence of study in the sections of the on-line course syllabus. (Note: the 6 & 8 week intersession cover the same material at a commensurate rate) Complete assignments and post those results requested.

Note: See Assignments section for complete description of each course requirement. You will earn points for completion of course requirements for the PROJECTS as described below:

CIS 17A

Grading Scale for **Programming** Elements of

Projects, Presentations

Assignments - On a scale of 1 to 10

9 - 10 points

- specifically addresses all parts of the Assignment
- thoughtful, clearly commented and well organized code
- demonstrates keen understanding of overall lesson content
- integrates content of recommended programming components
- shows relevancy to lesson content
- raises additional questions or issues not specifically covered in class
- includes justification and/or documentation to support results
- includes programming content from other Chapters not covered in class with full documentation flowcharts/doxygen.

7 - 8 points

- addresses Assignment in less specific terms
- may not always address all parts of the program
- code not always supported or justified

- demonstrates good understanding of overall lesson content
- may include digressions from lesson content
- some integration of information
- Minimal to inadequate flow-charting/doxygen.

5 - 6 points

- addresses Assignment general terms
- does not address all parts of the Assignment
- code frequently not supported or justified
- demonstrates some understanding of overall lesson content
- includes digressions from lesson content
- organization and focus are inconsistent

3 - 5 points

- organization and focus are poor
- does not address most parts of the Assignment
- code not supported or justified
- demonstrates minimal understanding of overall lesson content

Less than 2 points

- lacks organization and focus
- shows no understanding of lesson content
- fails to address any part of the Assignment
- no unique application of concepts

Grading

Your grade in this class will be based on your performance on assignments, projects, exams, and your participation in course learning activities. You will earn points for each course requirement completed; the number of points you earn determines the letter grade you receive according to the following scale:

Percentage	Grade
90 - 100	A
80 - 89	B
70 - 79	C

60 - 69	D
Below 60	F

Class Information Resources

If you find internet resources related to course content on your own, please share that information with the class by posting it on the Class Discussion Board.

COURSE STRUCTURE

How to Study

You should plan to devote at least 12 hours per week during a normal semester to complete the coursework/lesson for the week. For inter-sessions, this goes up dramatically to 24 to 36 hours a week. Establish a study schedule that works for you by setting aside a specific time to study when you are free of distractions and other commitments.

Preliminary to each lecture

Review the slides and the reading material in the text. Sets the stage for the lesson; contains the learning objectives and a summary of key points on which you should focus.

Reference material/Labs

This section may contain any combination of reading, listening, or viewing assignments and on-line resources for you to explore. Contains the background information you will need to explore in order to better understand the content.

Projects/concepts assigned.

Coding directly engages you in the learning process. You will be assessed to determine your mastery of the subject matter.

Discuss with fellow students your approach to solving problems.

Gives you the opportunity to "put it all together" through reflection and discussion with your classmates. You are expected to ask questions and/or answer questions should students post to the discussion boards.

Evaluate your solutions by trouble-shooting with various test-cases.

Assesses how well you implemented your assignments and projects.

ASSIGNMENTS

Suggested sequence of Study

Given there are roughly 16 lessons and 16 weeks to complete the course of study here is a suggested review sequence that will be useful each week:

Sunday	Review Text and Slides for the weeks lesson.
Monday- Tuesday	Book self-test/self-review/exercises.
Wednesday – Thursday	Attend Lab/Solve the problems.
Friday – Sunday	Post questions/answer questions.

Programming cannot be rushed. Each lesson takes 12 hours to properly cover, review and apply the material. It takes time to soak up i.e. absorb this information and even longer to connect it all to the big picture. Don't think that you are going to rush this to completion. If you are getting done sooner than expected, I would attempt more problems at the end of each chapter in the book and/or start planning to create a master project that is many more lines than the 1000+ required for the course. Practicing on all the problems in the book will make you an excellent programmer in the end.

Weekly Assignments

You will always write programs covering the lesson's content. The assignment will be submitted back to me for evaluation. Assignments must be turned in on time which will be no later than Sunday Evening at 11:59 PM of the week it is assigned. Once the time has expired, then turn in what you have and proceed to the next lesson. (Note: I always make the online assignments open 24 to 48 hours past this time in case there are server problems. I wouldn't take regular advantage of this.)

Mid-Term and Final Exams

These will be problems designed to test your knowledge of the material. Normally you will be given a little less than a week to complete the Exams. I want them run in Netbeans. You can choose to use other IDE's but Netbeans is what I will use. Fail to test them in Netbeans, and your grade could be impacted!

First and Final Project

These two projects are major assignments. There is a written part as well as a coded part. Just as in the homework assignments and exams, you will turn in files that I will run to evaluate your code. The first project is a minimum 250+ line program that you will write to solve a particular problem you are interested in. These are of your own choosing. The Final project is a minimum 1000+ line program that again is of your choosing but must solve a particular problem of interest. Examples will be referred to and suggestions will be given if you are having trouble determining what would be appropriate. Prior to working on the project, you will gain ideas from your classmates in the discussion board. You will also provide a proposal to turn in for my approval. A written report will be required when the projects are due as well as running code.