



The mission of Concord University is to provide quality, liberal arts based education, to foster scholarly and creative activities and to serve the regional community (<http://www.concord.edu/academics/>).

CS 202 – Introduction to Computer Programming II

20154, Section 01

Semester Taught: Spring 2019

Professor: Lonnie Bowe
(Mr. Bowe or Professor)

Credit Hours: 3

Office Location: Science 100E

Prerequisites: CS 201 with a grade of “C” or better

Office Hours: MWF 7:30 am – 8 am, 3pm-4pm

Course Time (if applicable): MWF 9:00 am – 9:50 am

TR 1:30pm – 2:00pm

Building and Room Number: Science 105

Other Times by Appointment

Email: lbowe@concord.edu

Phone: (304) 384-5387

Office Fax: (304) 384-6225

College/Department Website:
<http://math.concord.edu>

Course Description/Rationale: An introduction to object oriented programming concepts, fundamental data structures and algorithms, the construction of graphic user interfaces, and the software development life-cycle. Topics include: Classes, Inheritance, Polymorphism, Stacks, Queues, Linked Lists, Binary Trees, Hash Tables, Graphs, Searching and Sorting Algorithms.

Course Management System: <https://moodle.concord.edu>

Hardware/Software Needed: <https://www.concord.edu/technology/node/23>

Text requirements: Code Complete: A Practical Handbook of Software Construction, 2nd Edition
by Steve McConnell
ISBN 978-0735619678

Concord University Educational Goal(s):

- Skills:
- (1) Effective inter-communication skills and literacy adapted as needed for the demands of various kinds of discourse.
 - (2) An ability to employ appropriate observational, logical, analytical, computational, creative, and critical thinking skills within and across

academic disciplines; and to apply these skills in problem-solving.

National Standards: CS 202 comprises most of the topics recommended by the ACM in its 2013 Curriculum Guidelines for Undergraduate Degree Programs in Computer Science (See page 58 “Fundamental Data Structures and Algorithms” and page 167: “Software Development Fundamentals”). The ACM report can be found here: <http://www.acm.org/education/curricula-recommendations>

Specific Learning Outcomes:

1. Determine informally the time and space complexity of simple algorithms.
2. List and contrast standard complexity classes.
3. Implement simple search algorithms and explain the differences in their time complexities.
4. Be able to implement common quadratic and $O(N \log N)$ sorting algorithms.
5. Describe the implementation of hash tables, including collision avoidance and resolution.
6. Discuss the runtime and memory efficiency of principal algorithms for sorting, searching, and hashing.
7. Discuss factors other than computational efficiency that influence the choice of algorithms, such as programming time, maintainability, and the use of application-specific patterns in the input data.
8. Illustrate by example the basic terminology of graph theory, and some of the properties and special cases of each type of graph/tree.
9. Demonstrate different traversal methods for trees and graphs, including pre, post, and in-order traversal of trees.
10. Model a variety of real-world problems in computer science using appropriate forms of graphs and trees, such as representing a network topology or the organization of a hierarchical file system.
11. Explain why you might choose to develop a program in a type-safe language like Java, in contrast to an unsafe programming language like C/C++.
12. Describe the main concepts of the OO model such as object identity, type constructors, encapsulation, inheritance, polymorphism, and versioning.
13. Use subclassing to design simple class hierarchies that allow code to be reused for distinct subclasses.
14. Correctly reason about control flow in a program using dynamic dispatch.
15. Explain the relationship between object-oriented inheritance (code-sharing and overriding) and subtyping (the idea of a subtype being usable in a context that expects the supertype).
16. Use object-oriented encapsulation mechanisms such as interfaces and private members.
17. Define and use iterators and other operations on aggregates, including operations that take functions as arguments, in multiple programming languages, selecting the most natural idioms for each language.

18. For a language with a static type system, describe the operations that are forbidden statically, such as passing the wrong type of value to a function or method.
19. Describe examples of program errors detected by a type system.
20. For multiple programming languages, identify program properties checked statically and program properties checked dynamically.
21. Give an example program that does not type-check in a particular language and yet would have no error if run.
22. Use types and type-error messages to write and debug programs.
23. Explain how typing rules define the set of operations that are legal for a type.
24. Write down the type rules governing the use of a particular compound type.
25. Explain why undecidability requires type systems to conservatively approximate program behavior.
26. Define and use program pieces (such as functions, classes, methods) that use generic types, including for collections.
27. Discuss the differences among generics, subtyping, and overloading.
28. Explain multiple benefits and limitations of static typing in writing, maintaining, and debugging software.
29. Explain how programming language implementations typically organize memory into global data, text, heap, and stack sections and how features such as recursion and memory management map to this memory model.
30. Discuss the benefits and limitations of garbage collection, including the notion of reachability.
31. Identify the data components and behaviors of multiple abstract data types.
32. Implement a coherent abstract data type, with loose coupling between components and behaviors.
33. Compare alternative implementations of data structures with respect to performance.
34. Describe how references allow for objects to be accessed in multiple ways.
35. Compare and contrast the costs and benefits of dynamic and static data structure implementations.
36. Choose the appropriate data structure for modeling a given problem.
37. Explain why the creation of correct program components is important in the production of high-quality software.
38. Conduct a personal code review (focused on common coding errors) on a program component using a provided checklist.
39. Contribute to a small-team code review focused on component correctness.
40. Describe how a contract can be used to specify the behavior of a program component.

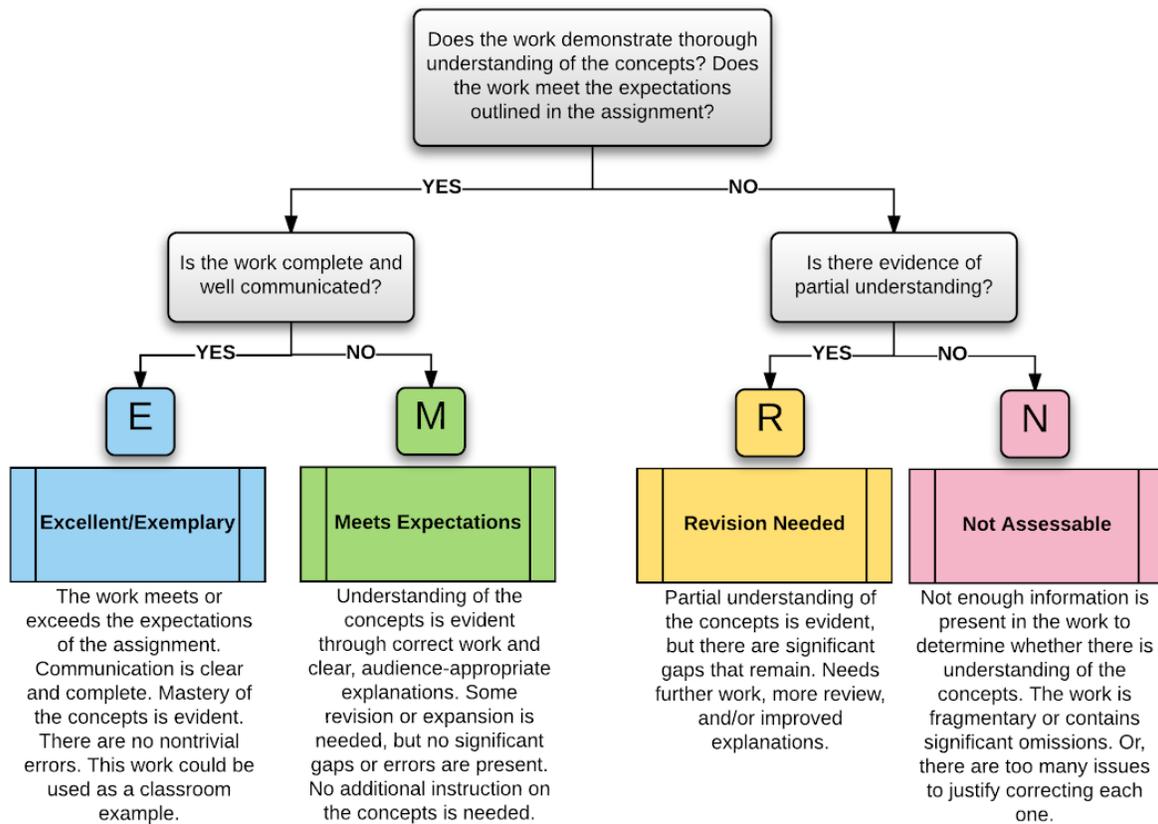
41. Construct, execute and debug programs using a modern IDE and associated tools such as unit testing tools and visual debuggers.
42. Construct and debug programs using the standard libraries available with a chosen programming language.
43. Analyze the extent to which another programmer's code meets documentation and programming style standards.
44. Apply consistent documentation and program style standards that contribute to the readability and maintainability of software.
45. Explain the concept of a software lifecycle and provide an example, illustrating its phases including the deliverables that are produced.
46. Describe how available static and dynamic test tools can be integrated into the software development environment.
47. List the key components of a data model (eg, class diagrams or ER diagrams).
48. Articulate design principles including separation of concerns, information hiding, coupling and cohesion, and encapsulation.
49. Describe techniques, coding idioms and mechanisms for implementing designs to achieve desired properties such as reliability, efficiency, and robustness.
50. Build robust code using exception handling mechanisms.
51. Select and use a defined coding standard in a small software project.

Course Requirements:

- 9 Mini-Projects:** To ensure that you have an understanding of syntax and concepts presented in class, there will be regular assignments for you to complete on an individual basis.
- 10 Exercises:** These short exercises are intended to reinforce and expand on lecture material.
- 15 Reading Responses:** Communication is an important but neglected skill for students. You will regularly receive assignments where you must read, explain or present new information.
- 14 Quizzes:** There will be short, weekly quizzes. You may have to design algorithms, read code, correct mistakes, and write small snippets of code.
- 1 Final Project:** To demonstrate what you have learned during the semester, you will be presented with a final project. The project will be assigned around mid-term and will be due during the finals time slot for this class. You will be required to demonstrate your project in front of the class and answer questions about it.

Grading Policy

All coursework will be graded on the EMRN system (<http://rtalbert.org/specs-grading-iteration-winner/>).



EMRN rubric based on the EMRF rubric, due to Rodney Stutzman and Kimberly Race: <http://eric.ed.gov/?id=EJ717675>

Each assignment will come with a revision policy and timeline.

The Final Exam will be graded as Pass / Fail.

Grading Scale

To Earn a D: You must earn a grade of M or better on 6 Exercises, 9 Reading Responses, 8 Quizzes, 5 Mini-Projects, and the Final Project.

To Earn a C: You must earn a grade of M or better on 7 Exercises, 11 Reading Responses, 10 Quizzes, 6 Mini-Projects, and the Final Project.

To Earn a B: You must earn a grade of M or better on 8 Exercises, 12 Reading Responses, 11 Quizzes, 7 Mini-Projects, and the Final Project.

To Earn an A: You must earn a grade of M or better on 9 Exercises, 14 Reading Responses, 13 Quizzes, 8 Mini-Projects, and the Final Project.

Late Assignments: Late assignment policies vary with the type of assignment. Each assignment will detail the late policy.

Missed Quizzes: You have 36 hours from the start time of the quiz to contact the instructor about a makeup quiz. You must have a valid reason for missing the quiz. Makeup

quizzes may have different questions covering the same content as the regular quiz. Makeup quizzes must be taken within four week days of the original quiz.

Grade Disputes: If you disagree with a grade or believe it is inaccurate, you may contest your grade within 7 calendar days from when the grade was released. After the 7 day **period, the grade becomes final.**

Attendance Policy: Attendance will be taken. You can miss three classes without penalty. After three, the instructor reserves the right to lower your grade at his discretion; after six absences you risk failure for the course.

Course Timeline: Students should expect at least one reading assignment per week, with frequent programming assignments. A full, tentative schedule will be posted to Moodle.

Attendance: You need to come to class. Attendance is encouraged. Participation exercises will be given regularly and are worth a portion of your final grade. Attendance and participation are important factors in all classes. This class, in particular, moves at a good pace and it is better for you to be there, even if you are half asleep. If you are sick, it is recommended that you stay home rather than infect the entire class. Participation exercises will often involve working in small groups and therefore cannot be completed at a later date. Note that this policy does not distinguish “excused” from “unexcused” absences – such a distinction puts me in a role I don’t want to play. University approved activities are exempt from this distinction. As always, exceptions will be made for extraordinary circumstances.

Attendance policy influenced by Brian Croxall, Emory University

The instructor follows the official University policy on student safety. Inclement weather conditions will be taken into consideration in regard to the attendance policy. <http://www.concord.edu/emergency-alerts>

Contact Policy: The instructor uses e-mail for class announcements. Please check your Concord e-mail at least once a day. You are responsible for the content of the e-mails.

The instructor is here to assist you. Remember that you can stop by during office hours, make an appointment, post on the course forum or send an e-mail! Please email from you MyCU email address.

When e-mailing the instructor, please allow for plenty of time to get a response as the instructor isn’t always online.

Feedback Policy: It is my goal to give you timely, constructive feedback on your performance in this class. However, I cannot know how I am performing without feedback from you. I encourage you to let me know how I am doing, what you like and dislike about the course.

This feedback will in no way impact your grade. Do not suffer in silence. If speaking to me in person makes you uncomfortable, I will also periodically provided anonymous surveys on Moodle for you to express your opinions.

Cell Phone, Tablet, Laptop, work from other classes, etc. Policy: It is best if you are engaged in the class, if I feel that you are not, then I reserve the right to remove you from the classroom.

Other policies may be distributed in class or on Moodle. Students will be held responsible for these policies.

Mandatory Syllabus Statements that apply to all CU courses:

Accessibility/Accommodations:

Concord University is committed to responding to the needs of students with disabilities as defined by the Americans with Disabilities Act. Please inform your instructor at the beginning of the class semester if you have a disability and are requesting accommodations. It is your responsibility to self-disclose that you are requesting accommodations. The University and instructor will provide you with a reasonable accommodation. You should register with CU's Disability Services Office, located in the Athens campus Jerry and Jean Beasley Student Center, Bottom Floor, across from the Campus Post Office. The Disability Services Office phone is 304-384-6086 or you can email the Director, Nancy Ellison, at nellison@concord.edu for assistance.

Academic Dishonesty

Academic dishonesty is morally unacceptable as well as destructive to the learning and teaching atmosphere. Academic dishonesty includes the giving or receiving of improper help on examinations or assignments, falsifying documents, and plagiarism (the act of stealing and using, as one's own, the ideas or the expression of the ideas of another). Such dishonesty can lead to a variety of penalties — including but not limited to failure of assignment, failure of course, loss of institutional privileges, or dismissal from the University. (See University Catalog Academic Policies and Procedures.)

Concord University Honor Code

A Concord University Honor Code was approved by students, staff, faculty, administration, and the CU Board of Governors. The Code states:

"As a member of the Concord University Community I will act with honesty and integrity in accordance with our fundamental principles and I will respect myself and others while challenging them to do the same."

The Honor Code is intended to unite the Concord community behind a culture of honesty, integrity, and civility.

Class/Online Attendance Policy

Regular class attendance is part of a student's academic obligation at Concord. Irregular attendance may affect academic performance adversely and is detrimental to the atmosphere of a class. (See University Catalog Academic Policies and Procedures.)

Emergency Alert System

In an effort to increase safety and security on our campus, Concord University encourages everyone to register for instant text message alerts. Alerts will only be used for security and safety notices. All students, faculty, and staff are eligible to receive text message alerts on their cell phones or email alerts. Please contact the IT Help Desk for further assistance (304-384-5291).

Emergency Information

Emergency/courtesy telephones are located at the main entrance of each residence hall and at various other locations on campus. Emergency telephones can be identified by the flashing blue light and will provide the user with a direct link to Public Safety at the press of a button. To report an on-campus emergency, call 304-384-5357 or 911. The Office of Public Safety is located on the bottom floor of the

Rahall Technology Center. For further emergency information go to:
<http://www.concord.edu/administration/office-public-safety>.

Inclement Weather Policy

As a general policy, the University will remain in normal operations during adverse weather conditions. In the event of severe weather conditions, the following may occur:

University Closure

No students or employees are to report.

Classes Cancelled

Students do NOT report BUT employees are expected to report to work at their normal time.

Operating on an Inclement Weather Delay

Under this schedule, all 8 a.m. classes will start at 10 a.m. Students and faculty will follow the Inclement Weather Schedule. (See <http://www.concord.edu/emergency-alerts> for Athens/Beckley Inclement Weather Schedules.)

**Announcements invoking the late schedule or other options referenced above are aired on area radio and television stations and are sent as text and email messages to those enrolled for this service.*

Student Conduct

In classrooms, online, laboratories, and during any activities that are part of course requirements, students are expected to observe reasonable rules of conduct.

Sexual Harassment & Assault

Federal law, Title IX, and Concord University policy prohibits discrimination, harassment, and violence based on sex and gender (Including sexual harassment, sexual assault, domestic/dating violence, stalking, sexual exploitation, and retaliation). If you or someone you know has been harassed or assaulted, you can receive confidential counseling support through the Concord University Counseling Center (304-384-5290). Alleged Violations can be reported non-confidentially to the Concord University Title IX Coordinator at 304-384-6327 or titleix@concord.edu. Reports to Campus Security can be made at (304-384-5357). As an employee at Concord University, I am a mandatory reporter which means I must report any sexual misconduct I am made aware of. This includes verbal or written (such as in an assignment) disclosures of sexual harassment or sexual assault.

Technology Services

Contact the CU Help Desk at extension 5291 from campus or 304-384-5291 off campus. You may also e-mail cuhelpdesk@concord.edu.

Syllabus Disclaimer

"This syllabus is subject to change based on the needs of the class. Please check it regularly."