

Syllabus – CSC 161 – section 2 – Spring 2019

CSC 161 is the second course in Grinnell's [3-course, introductory computer science sequence](#)

The course explores elements of computing that have reasonably close ties to the architecture of computers, compilers, and operating systems. The course takes an imperative view of problem-solving, supported by programming in the C programming language. Some topics include:

- *imperative problem solving*: top-down design, common algorithms, assertions, invariants
- *C programming*: syntax and semantics, control structures, functions, parameters, macro processing, compiling, linking, program organization
- *concepts with data*: data abstraction, integer and floating-point representation, string representation, arrays, unions, structures, linked list data structures, stacks, and queues
- *machine-level issues*: data representation, pointers, memory management
- *GNU/Linux operating system*: commands, bash scripts, software development tools

Course Work

Course Format and Activities

This course follows a lab-based format. Thus, work related to [almost] all class sessions proceed in three stages:

1. **Read Before Class:** Reading assignments will incorporate both assigned readings from the textbook and online readings linked from each module. Textbook readings are assigned on most days, and those assignments can be found noted on the schedule. Online readings are linked from the module(s) for most days. Students are expected to complete assigned readings *prior to each class session*. Each student should prepare a list of questions or topics for class discussion, should questions arise on the reading.
2. **Ask Questions and Work on Lab during Class:** Each class will begin with questions from students, based on the reading. Following this initial discussion, the course may involve some opening comments or a short quiz on the reading. After these preliminaries, students will work collaboratively, in pairs, on a lab exercise based on the reading.

3. **Finish Lab for Homework:** Although the in-class labs will help students get started on the lab exercises, students may not complete the lab during the class period. Any lab work not completed during class should plan be finished as part of homework. I will assign lab partners, with pairings changed approximately every week.

In addition to an in-class, lab-based format throughout the semester, additional homework is assigned to provide additional practice and to suggest a range of applications beyond the course's robotic application theme.

Course Components

Course Work will involve a combination of several activities. [Submission instructions](#) vary somewhat according to the type of the assignment.

- [Homework Assignments](#) extend the range of problems considered in the course and help sharpen problem-solving skills. Through the semester, **four** homework assignments will be required.
- [Module Laboratories and Projects:](#) Topics in this course are organized into eight modules and a few supplementary labs. Modules include laboratory sessions and conclude with a integrative project.

Laboratory sessions introduce specific features of Grinnell's computing environment, highlight concepts and constructs introduced in class, allow instructor assistance in a "hands-on" setting, and supplement normal office hours. Approximately five labs require a formal write-up, explaining what work you have done, showing any programming you have done, indicating tests or experiments run, and giving your conclusions. Labs designated [*Required*] on the [Class Schedule](#) are required -- labs designated [*Extra Credit*] may be done for extra credit. In order for extra-credit labs to provide reasonable benefit, any such lab being submitted for credit must be turned in within 1 week of when it is scheduled for coverage in class. Extra credit may not raise a student's average on laboratories above 120%.

Note: As a special incentive for mastering the laboratory exercises, *between a third and a half of the problems on each test and on the exam will be taken from the laboratory exercises (with only slight editing).*

Projects: Most of the eight modules conclude with a project, involving both a computer program and associated commentary, and each project is a required part of the course.

Collaboration on labs and projects encouraged: Many lab activities and projects should be done in groups of two or three, although you may work individually on labs after Lab 8. Only one write-up is expected per group.

- **Quizzes:** At the beginning of some class sessions (after initial questions/answer period), a short (5-10 minute) quiz may cover basic topics from the reading(s) for the day. There will be 8 quizzes, and the lowest score will be dropped from grade calculation. Questions for the quiz will be drawn from both the textbook and online readings.
- **Mid-Term Exams:** There will be two "mid-term" exams during the course.
- **Exam:** An in-class final exam will be scheduled for this course. Following the exam schedule published by the Registrar's Office, times for the exam are as follows:
 - Section 161.01: TBD
 - Section 161.02: TBD

This formal schedule notwithstanding, students in either Section 161.01 or Section 161.02 will be allowed to take the exam at the times scheduled for either section — following procedures prescribed by the College. Details will be discussed in class.

- **Extra Credit Opportunities:** This course provides numerous ways for earning extra credit.
 - *Tuesday Tables and Thursday Extras:* Computer science is a wide-ranging discipline, and courses can cover only selected pieces. To encourage students to expand their horizons, students may earn 2 points extra credit for each *Thursday Extra* or other departmental talk, by attending the talk and writing a short (4-8 sentence) summary or response. Logistically, a reasonable statement must be submitted via e-mail to the instructor **within 1 week** of the talk; after the e-mail has been reviewed and the statement deemed appropriate, the student will receive 2 extra credit points that will apply to the laboratory category.
 - *Labs:* Some laboratory exercises on the schedule are designated as being available for extra credit. To earn points, turn in a lab write up within 1 week of when it is covered in class. As with Supplemental Problems, extra credit may not raise a student's average on assignments above 120%.
 - *Additional extra credit opportunities* may be announced through the semester.

Anticipated Work Load

As noted above, this course includes a variety of activities, including class preparation (reading), labs and projects (started in class and finished for homework), and homework problems, as well as quizzes, tests, and a final exam. From past experience, the time required for these activities will likely vary substantially from student to student and from one part of the course to another. For example, a student may need to devote considerable time and effort when starting a new or different topic, but the workload may drop noticeably when that material is mastered.

Such variation in student experiences complicates any estimation of the time individual students may need to devote to homework for this course. However, from past experience, students working steadily on the course likely should expect to allocate 10-15 hours per week to homework. Some students may require additional time for some weeks; some students may complete work in less time for some weeks. Conversations with computer science faculty and others suggest this time allocation is consistent with expectations for many courses at Grinnell College.

Resources

Instructor

[Barbara Z. Johnson](#)

Office: Science 2811

Telephone: extension 4695

E-mail: johnsonba@grinnell.edu

I will do my best to respond to email or telephone calls within 24 hours during the week. I will probably not be checking email over the weekend.

Office hours will be announced during the first week of class.

If the door is closed, please assume that I am meeting with someone. If none of these times work for you, please check [my calendar](#) for an available time and email me to set up an appointment.

Reference Materials

Required Textbook

K. N. King, [C Programming: A Modern Approach, Second Edition](#), W. W. Norton, 2008, ISBN 978-0393979503.

King often covers details that are not in the online readings, especially regarding the evolution of the language. As an incentive to read the assigned readings, approximately half of the quiz questions will come from King, with minor variations.

Readings and examples for this course are under development. As noted in the [day-by-day schedule](#), complete readings, examples, projects, and labs are available for most class sessions throughout the semester. The materials available for this course extend beyond 100,000 lines of text. There are some typos, but I am doing my best to correct them as they are located.

Additional References

- If you want a more compact reference book, I still carry around my copy of: Brian W. Kernighan and Dennis M. Ritchie, [The C Programming Language, Second Edition](#), Prentice Hall, 1988, ISBN 0-13-110362-8 (paperback), 0-13-110370-9 (hardback).
- Henry Walker, *An Introduction to C Through Annotated Examples*, <http://www.cs.grinnell.edu/~walker/c/index.html>
- [The GNU make Manual](#), Free Software Foundation, 2006.
- Eric Huss, [The C Library Reference Guide](#), University of Illinois Student Chapter, 1997.
- The [MyroC documentation](#) that identifies MyroC operations defined for the Scribbler 2 robots in C.
 - [MyroC data structures \(Picture, Pixel\)](#)
 - [MyroC header file \(Web-based format\)](#)
 - [MyroC header file \(C-style format\)](#)
- Musical scale data: Frequencies of notes in a well-tempered Western scale may be specified either as separate notes or using a `struct` data structure.
 - Code for the piano may be found in the header file [scale-notes.h](#)
 - Use of a `struct` data structure is described in the [reading on structs](#).
Code involves a [pitch.h](#) file to define the `struct` framework and a [scale.c](#) file that specifies the pitches.
- [A diagram of Scribbler 2 sensors and function calls](#).
- Sound is available through headphones through most MathLAN workstations. However, sometimes the audio setting is muted. [Instructions for sound](#) are available, in case adjustments are needed.
- The eSpeak package that identifies speech-synthesis capabilities for use in C programs. We will not use eSpeak extensively, but you may find it helpful.
 - [eSpeakPackage.h](#) header file

- [Introduction to using the eSpeak package within C, through the use of the eSpeakPackage.](#)
- [Sample program eSpeakExample1.c](#)
- [Sample program eSpeakExample2.c](#)

A Note on Typos

Planning and development of materials for this course represent an extensive effort:

- Preparation and refinement of the course's structure and materials exceed 4100 hours.
- Course materials include about 270 Web pages involving over 100,000 lines of text (in HTML with php).
- Sample C code includes about 265 programs.

Although the developers have read, re-read, and refined the materials extensively, one can be confident that typographical errors remain.

If you find an error, if something does not read well, if deadlines on one page do not seem to match those stated on another page, etc. — don't panic (or use colorful language). Rather, please talk to the instructor (nicely please). Thanks!

Collaboration

Pair Responsibilities

Work on labs and projects in this course is often done collaboratively (in pairs, occasionally in a group of three). Many studies suggest substantial benefits to learning with this type of group work, and it is an industry practice in some software development methodologies. However, to be successful, collaboration requires partners to actively participate.

- Each partner should come to class and actively participate throughout the class session.
- Partners should make arrangements to meet as needed in the lab outside of class to finish labs or projects.
- Each partner has an obligation to show up and actively participate during out-of-class, planned sessions.

Failure to meet one's responsibilities to a group not only impacts the individual, but also impedes the education of the partner. Thus, except in exceptional circumstances (e.g., illness, family emergencies, serious injury),

failure to follow through with one's responsibilities as a partner may have a significant impact on one's course grade and/or one's standing in the course. Possible ramifications of repeated absences include receiving only partial credit for a project or lab submission or a reduction in overall course grade. See [Responsibilities](#) for pair programming for details.

Academic Honesty

All work in this course is governed by the rules of the college regarding academic honesty. In summary, standard practice requires that you must acknowledge all ideas from others.

- When working on homework, either individually or in a group, you may use any written source. However, the normal rules of citation must be followed, as described in the *Student Handbook*.
 - You must cite statements from the online course materials, if you use them to guide your work.
 - Although the World Wide Web can be useful for reference, you are advised that much material on the Web is of poor quality.
 - You are responsible for the quality of what you turn in, regardless of the source of the material.

- When you work in a group on an activity, the names of all students in the group should appear at the top of the first page. Turning in work with multiple people listed as authors implies that all members of the group agree with what is presented. If a group member does not agree with some part of the work, the group should continue to discuss and revise the material until agreement is achieved. In summary, a group activity is a joint effort, and all group members have equal responsibility for the finished product.

- When you work on an activity yourself, but consult others, then you should include a statement identifies whom you consulted on what material. This includes conversations with class members, tutors, lab assistants, other students, faculty, and any other people involved. If you consulted one person on several parts on an assignment, you may summarize the collective help in an "Acknowledgment Section" rather on the individual problems — as long as the work makes clear who helped on what. Overall, all consultations on problems must be cited for each problem or exercise.

Collaboration

Collaboration is allowed on laboratory exercises and projects (i.e., all work done in class), but collaboration normally will NOT be allowed on homework problems, quizzes, and tests.

Course Policies

Dates, Deadlines, and Attendance

I do not take attendance, but since you work in pairs, I will generally know if you are missing from class. Repeated unexcused absences will impact your grade; so let me know when you plan not to come to class for any reason.

Grinnell College offers alternative options to complete academic work for students who reserve religious holy days. Please contact me **within the first three weeks** of the semester if you would like to discuss a specific instance that applies to you. Since this class relies on paired programming, it is important that I know you will be absent ahead of time so that I can adjust the pair assignments!

Normally, a laboratory write-up, project, or program is due about every third class meeting. Projects, and homework problems all require work to be submitted via email to `csc161-01-grader@grinnell.edu` or `csc161-02-grader@grinnell.edu` and in paper form. (The email address should correspond to your CSC 161 section.) Both email and paper copies must be received before the stated deadline. Laboratory exercises do not need to be emailed - just hand in your paper copy in class when it is due.

- The time stamp on email to `csc161-01-grader@grinnell.edu` or `csc161-02-grader@grinnell.edu` will be checked for timeliness.
- Paper must be turned in under my door ahead of time or turned in as designated in class.

It is expected that the materials submitted electronically will match the materials submitted in paper form. Discrepancies may raise questions of academic dishonesty and be subject to review by the Committee on Academic Standing.

Deadlines are shown on the [Class Schedule](#) , and work is due at the start of each class specified. A penalty of 10% per day (weekends count as one day) will be assessed for any assignment turned in late, even work submitted in the middle or the end of a class. I generally allow 5 to 10 minutes leeway for printing and getting your lab partner to sign the Academic Honesty section,

but more than 15 minutes after the start of class will result in a reduction of points. Print your materials well before the start of class!

Exceptions to the deadline policy and its penalties:

- An extension of at least one class period is automatically granted if the department's Linux network is down for an unscheduled period of three or more hours during the week preceding the assignment.
- Difficulties sometimes arise with printing through PaperCut. When such difficulties arise, you should contact the instructor (either at the start of class or via email). Note, however, that difficulties with PaperCut do not impact deadlines for emailing materials.
- Students arriving late to a class should turn in the paper form of the work as soon as they arrive in the classroom. In such cases, the assignment will be considered on time if (and only if) the email submission was received before the class began.

Emergencies, Illness

Although dates for labs, programming assignments, tests, and the final exam are firm, I understand that circumstances arise when you are not able to attend class.

- When circumstances are known ahead of time (e.g., academic activities, athletic events, religious holidays), I expect you to make arrangements with me before the activity occurs. Normally, we will identify an alternative date for the due date or test. And I may adjust paired programming assignments to team you with someone else who will be absent that week.
- When circumstances cannot be reasonably anticipated (e.g., illness, family emergencies, serious injury), I expect you to notify me as soon as is reasonably possible. (Email is fine.) In the case of medical problems that require you to miss several classes, I expect a written note from a medical professional or counselor that indicates that your health interfered with the course activity. (I do not need to know any details of the medical problem, but I do need to know that you sought help and that the medical professional believed meeting the deadline would likely interfere with your health.)

Absolute Deadline: *All homework must be turned in by the last day of classes (May 10) at 5:00 pm; laboratory reports, projects, or programs received after that time will not be counted in the grading of the course.*

Cell Phones, Text Messaging, and E-Community Devices

Cell phones, text-messaging devices, and other social-networking connections may not be used in this class. If you bring such equipment to the classroom, it must be turned off before the class starts and stay off throughout the class period. Use of such equipment is distracting to those nearby and will not be tolerated. Also, do not use your laptops or iPads to watch videos unrelated to the class material - I will ask you to put them away.

Accommodations

If you have specific physical, psychiatric, or learning disabilities and require accommodations, please let me know early in the semester so that your learning needs may be appropriately met. You will need to provide documentation of your disability to the Office of Student Affairs. Feel free to talk to me if you have questions or want more information.

Grading

This instructor's grading philosophy dictates that the final grade should ultimately be based upon each student's demonstration of her or his understanding of the material, not on the performance of the class as a whole nor on a strict percentile basis. The following scheme is proposed as a base for how the various assignments and tests will be counted in the final grade.

Lab Write-ups:	Homework:	Projects:	Quizzes:	Mid-Terms :	Final Exam:
15%	20%	20%	5%	20%	20%