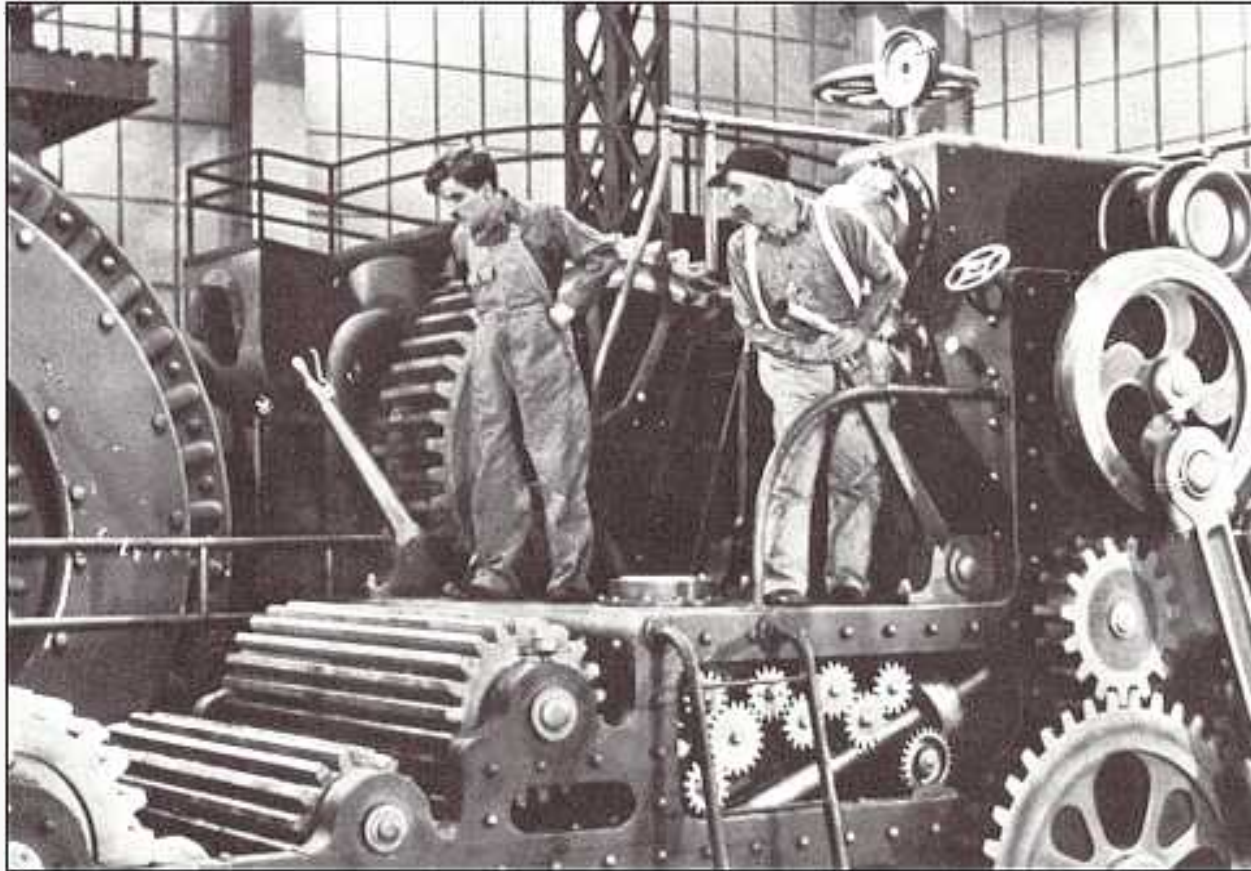


Computer Science 2505
Computer Organization I



<http://courses.cs.vt.edu/cs2505/fall2020/>

Modern Times

Chaplin

William D McQuain

Email: `wmcquain@cs.vt.edu`
Zoom: `virginiatech.zoom.us/my/wmcquain`
Office: 634 McBryde Hall (N/A for Fall 2020)
Office Hours: see course website

The Plan

- lectures will be recorded and posted on Canvas
- all other course materials will be posted on the main course website
- all of my office hours will be held on Zoom
- some office hours will be *en masse*, like a help session
- some office hours will be one-at-a-time
- tests will be offered via Canvas, using the Lockdown Browser and Respondus Monitor
- I will NOT be available for in-person consultations

<http://courses.cs.vt.edu/cs2505/fall2020/>

<https://canvas.vt.edu/courses/115595/>

David McPherson

Email: `dmcphers@vt.edu`

Zoom: `https://virginiatech.zoom.us/j/94068175391`

Office: 631 McBryde Hall

Office Hours: see course website

<http://courses.cs.vt.edu/cs2505/fall2020/>

<https://canvas.vt.edu/courses/115595/>

An introduction to high-level computer architecture and systems concepts, system software and programming in C.

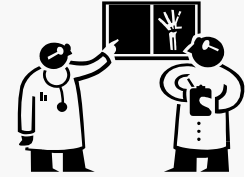
Covers information representation, basic computer organization, fundamentals of the x86 architecture, the relationship between the C programming language and x86 assembly language, the use of debugging and other system tools, and the development of small- to medium-sized software solutions in C.

The course will focus on the x86-64 architecture.

Having successfully completed this course, the student will be able to:

- understand the representation of information at the hardware level,
- understand the different layers of abstraction in a computing system, i.e., logic design, computer architecture, machine language, assembly language, high level language,
- write simple to moderately complex programs in C,
- read and understand simple programs in x86 assembly language.
- be able to apply common UNIX tools, such as gdb and objdump, to analyze, debug and correct C programs,
- be able to operate efficiently in a UNIX command-line environment.

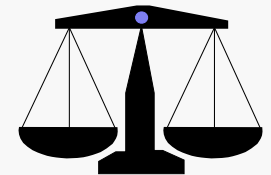
Final grades will be based on the average achieved over the following :



Item	Weight	Dates
Assignments	60%	See course website
Tests (2)	10% each	October 9 and November 11
Final Exam*	20%	December 13 details TBA

Grade Scale

The usual 10-point scale will apply (subject to any curve). A final average of 90% will guarantee an A-, 80% will guarantee a B-, and so forth.



Curve

A grade curve may or may not be employed in this course. The application of a curve is dependent upon class performance on tests, projects and homework. The decision to utilize a curve rests entirely with the course instructors.

* Exam score will replace the lower test score, if it is higher.

Lectures

PowerPoint course notes will be posted on the CS-hosted website (not Canvas).

I will make one (or more) lecture recordings for each set of notes.

You should review each set of PowerPoint notes before viewing the related lecture recording(s).

The lecture recordings will be posted on the course Canvas site.

Assignments

You will submit all of your assignment solutions via the Curator system.

Non-programming assignments will be graded manually by the course TAs, and feedback will be provided via the Curator.

Programming assignments will be evaluated by using stand-alone automated grading packages, which will be supplied to you.

Office Hours

I will not be holding any "live" office hours.

Instead, I will host my office hours on Zoom; my plan is that:

- some office hours will be restricted to a particular course (I'm also teaching 3114)
- some office hours will use the Zoom waiting room; during those I will usually interact with one student at a time
- some office hours may be mass meetings
- see the CS 2505 Forum board for announcements

Help Sessions

I will also host occasional help sessions on Zoom:

- each of these will be devoted to a particular assignment, and will be "open" mass meetings
- I hope to incorporate some Q/A interactions in these sessions
- details will be announced on the CS 2505 Forum board

Administering Tests

Obviously, the two tests and the final examination will be given online.

I anticipate that each test:

- will be hosted on Canvas
- will require your use of the Lockdown Browser with Respondus Monitor

Instructions for this will be supplied in advance of each test.

CS 2114 Software Design and Data Structures

or

ECE 2574 Data Structures and Algorithms

The prerequisite must have been completed with a grade of C or higher (C- is not acceptable) or AP credit.

Math 2534 Discrete Mathematics

or

Math 3034 Intro to Proofs

Must be taken before or concurrently with CS 2505.

We will not grant any exceptions to the stated prerequisites, including the minimum grade requirements.

Melissa Cameron	melissacameron	G
Bowen Shen	bowenshe	T
Ola Karajeh	okarajeh	A
Emma Meno	emmam99	s
Allyson Senger	asenger	U
Cameron Brown	bcameron19	T
Joseph McAlister	josephrm	A
Sai Krishna Yeshala	saikrishna	s

All TA office hours will be held on Zoom.

The course website includes a link to a Google Calendar showing those hours.

Office hour changes will be reflected on that calendar and announced on the course Forum Board.

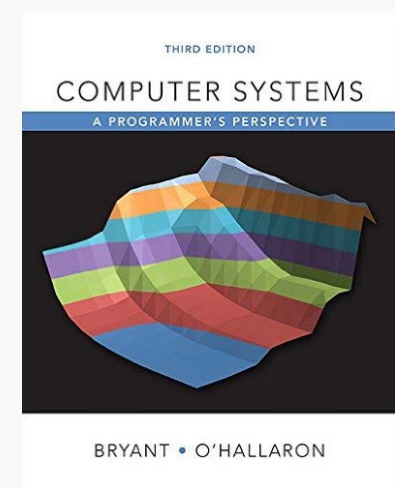
Required:

Computer Systems: a Programmer's Perspective, 3rd Edition

Randal E Bryant & David R O'Hallaron

Addison Wesley ©2016

ISBN 978-0-13-409266-9



Recommended:

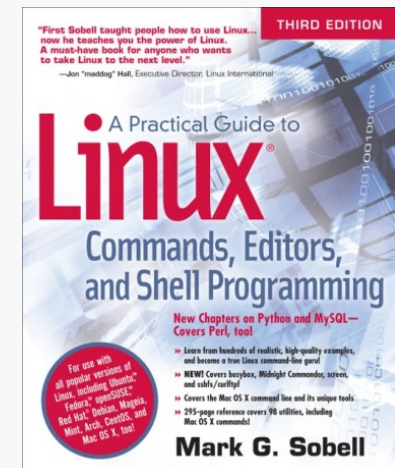
A Practical Guide to Linux Commands, Editors, and Shell Programming, 2nd, 3rd or 4th Ed.

Mark G Sobell

Prentice Hall ©2017

ISBN 978-0134774602 (4th Ed.)

(available via the Safari Database in the VT Library)

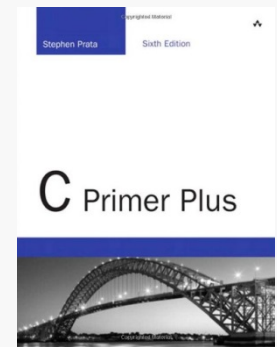


C Primer Plus, 6th Edition

Stephen Prata

Developer's Library ©2014

978-0-321-92842-9

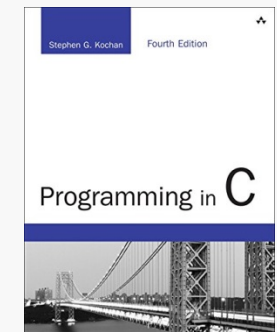


Programming in C, 4th Edition

Stephen G Kochan

Developer's Library ©2014

978-0-321-77641-9



CS 2505 Course Notes, Fall 2020 Edition, W D McQuain, ©2005-20



Available via the Safari Database in the VT Library:

The Art of Debugging with GDB, DDD, and Eclipse

N Matloff & P J Salzman, No Starch Press ©2008 ISBN 978-1-593-27174-9

Write Great Code, Volume 1

R Hyde, No Starch Press ©2004 ISBN 978-1-593-27003-2

Write Great Code, Volume 2

R Hyde, No Starch Press ©2006 ISBN 978-1-593-27065-0

In order to receive a grade of C or higher for the course, you are required to meet some minimum requirements:

Binary bomb You must defuse at least the first four phases

Understand something clearly: if you fail to meet the stated requirements, you will receive a grade of C- or lower for the course, regardless of your overall average.

Each assignment will specify certain requirements:

- exactly what is to be turned in (for example, which C files)
- the format of the files to be turned in (for example, plain text, tar file, etc.)
- the time by which solutions must be turned in for full credit
- whether solutions will be accepted after that time

It's your responsibility to make sure that you turn in the correct thing(s), in the specified format, and by the specified deadline.

It's our responsibility to accurately evaluate what you turn in.

There are many students, and a small course staff.

Therefore, we will not provide any special treatment in cases where, due to student inattention, the wrong thing is submitted, or the submission is in the wrong format, or the submission is late.

You will probably be allowed to work together on some assignments. If you are allowed to work in pairs, or groups, it is important you understand what we expect.

Acceptable pairs work requires:

- Each partner contributes to the analysis of the assignment, and to the derivation of a solution. This does not mean the contributions will always be equal, but both partners must be actively involved. When the solution is complete, each partner should understand the entire solution.
- The partners do not "divide" different parts of a multi-part assignment, with each working independently on his/her parts and having little or no involvement in the other parts.
- No partner "runs away" with the problem and solves it independently, not giving the other partner(s) the opportunity to contribute.

If you are allowed to work in pairs, you will choose your partner. Choose wisely.

We will not make any accommodations for situations in which one partner makes an error in a submission; it is the responsibility of both partners to make sure everything is correct.

Test Environments

- When relevant, a test environment, will be specified for homework assignments.
- For programming assignments, testing will be done on CentOS 8.
- The C-language assignments will be compiled with gcc 8.3 (as installed on the rlogin cluster).
- Solutions will only be tested under the specified environment.
- It is the **YOUR** responsibility to ensure that **YOUR** solutions execute correctly in the appropriate environment; solutions that do not will receive substantial deductions.



The rlogin cluster is a collection of virtual machines running CentOS Linux, provided for the use of students in certain CS courses.

Some assignments may require you to work on the rlogin cluster.

Therefore you are required to run CentOS 8 Linux on your own computer, so that you have a stable development environment for your personal use in this course.

Notes explaining how to set up a CentOS Linux installation will be supplied on the Resources page of the course website.

Due dates

Each programming project and homework assignment will have a due date and time and will include instructions for submission.

Homework

Usually, no late submissions will be allowed for homework assignments.

Programming Assignments

Except in the very rare case that an extension is granted, late submissions will incur a penalty per diem late penalty that will be included in the project specification. This is typically 10%.

Extensions

Any request for an extension must be made, preferably by email, at least 24 hours prior to the due date.

Late submissions will not be given any credit if submitted after graded assignments or solutions have been released.

Statute of Limitations

Any questions about the grading of an assignment must be raised with your instructor within two weeks after the graded assignment has been made available to you.

This policy is motivated by several considerations:

If we have graded something incorrectly, that needs to be fixed for all the students who might have been involved, and that should be done as soon as possible.

If we have deducted points for a valid reason, you need to understand why your answer was inadequate, and you need to understand that before your confusion leads to errors on later assignments.

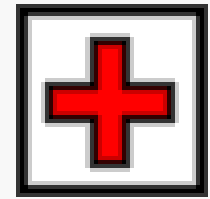
This policy is NOT intended to provide you with an opportunity to fix your errors after an assignment has been graded.

Our obligation is to correctly evaluate what you submit for grading.

Your obligation is to submit correct work, in the correct format, by the specified deadline.

General Issues

- CS 2505 classmates
- CS 2505 Forum board (forum.cs.vt.edu)
- CS 2505 TAs
- CS 2505 Instructor

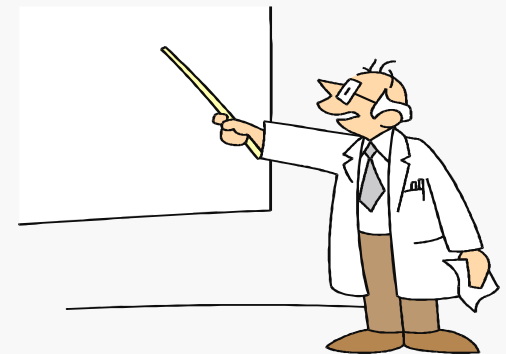


Programming Language Help

- B&O text and other resources from the course website
- C language references
- CS 2505 Forum board

Lecture Instruction

Lectures will consist of presentations, applications, problems and solutions interspersed with classroom discussion.



Backups

Students are responsible for making backup copies of all their work in this (and all) courses.

Loss of work due to hard drive failure is **NOT** an acceptable excuse. Backup copies of files on the same hard drive are not backup copies. Backup copies of files on second hard drives are also risky. Backup copies should be maintained on two separate distinct storage mediums, (e.g., hard drives and Zip disks).

Backup copies should be maintained until after the end of the term and students have received their course grade. (The Army lives by triplicate for a reason.)

Remember: Computer systems are mechanical devices.

Systems fail. Drives die. Bad sectors appear.

Network connections break.

Plan for it. It is inevitable!



The Undergraduate Honor Code pledge that each member of the university community agrees to abide by states:

“As a Hokie, I will conduct myself with honor and integrity at all times. I will not lie, cheat, or steal, nor will I accept the actions of those who do.”

Students enrolled in this course are responsible for abiding by the Honor Code.

A student who has doubts about how the Honor Code applies to any assignment is responsible for obtaining specific guidance from the course instructor before submitting the assignment for evaluation.

Ignorance of the rules does not exclude any member of the University community from the requirements and expectations of the Honor Code.

For additional information about the Honor Code, please visit:

<https://www.honorsystem.vt.edu/>

Commission of any of the following acts shall constitute academic misconduct. This listing is not, however, exclusive of other acts that may reasonably be said to constitute academic misconduct. Clarification is provided for each definition with some examples of prohibited behaviors in the Undergraduate Honor Code Manual located at <https://www.honorsystem.vt.edu/>

CHEATING

Cheating includes the intentional use of unauthorized materials, information, notes, study aids or other devices or materials in any academic exercise, or attempts thereof.

PLAGIARISM

Plagiarism includes the copying of the language, structure, programming, computer code, ideas, and/or thoughts of another and passing off the same as one's own original work, or attempts thereof.

FALSIFICATION

Falsification includes the statement of any untruth, either verbally or in writing, with respect to any element of one's academic work, or attempts thereof.

FABRICATION

Fabrication includes making up data and results, and recording or reporting them, or submitting fabricated documents, or attempts thereof.

MULTIPLE SUBMISSION

Multiple submission involves the submission for credit—without authorization of the instructor receiving the work—of substantial portions of any work (including oral reports) previously submitted for credit at any academic institution, or attempts thereof.

COMPLICITY

Complicity includes intentionally helping another to engage in an act of academic misconduct, or attempts thereof.

VIOLATION OF UNIVERSITY, COLLEGE, DEPARTMENTAL, PROGRAM, COURSE, OR FACULTY RULES

The violation of any University, College, Departmental, Program, Course, or Faculty Rules relating to academic matters that may lead to an unfair academic advantage by the student violating the rule(s).

An exhaustive list of Honor Code violations would be impossible to present here, but among other things, each of the following is a flagrant violation of the Virginia Tech Honor Code, and violations will be dealt with severely (Honor Court):

- Working with another student to derive a common program or solution to a problem. **Unless explicitly stated otherwise, there are no group assignments in this course.**
- Discussing the details required to solve an assignment. You may not share solutions, or collaborate in the creation of a solution.
- Copying source code (programs) in whole or in part from someone else.
- Copying files from another student's disk or lab account even though they might be unprotected.
- Editing (computer generated) output to achieve apparently correct results.

It is acceptable to discuss an assignment with classmates in a general way, i.e., to discuss the nature of the assignment. In other words, you may discuss with your classmates what your solution is required to accomplish but not how to achieve that goal using C, MIPS32 assembly, or other relevant tools. In no way should the individual statements of a program or the steps leading to the solution of the problem be discussed with or shown to anyone except those people cited in the following statement.

Feel free to discuss the homework assignments and your program source code with the teaching assistants assigned to your CS 2505, and the course instructors. The discussion of your program source code must be limited to these people. Note that this specifically excludes discussions of your program source code with other students (even if they are not enrolled in CS 2505), or with tutors. Privately hired tutors are not an exception to this requirement, nor are athletic or other tutors provided by the University.

Copies of all submitted work are retained indefinitely by the Department. Submitted programs are subjected to automated analysis for detection of cheating.

If you have any question as to how the Honor Code applies to this class, remember that:

- Any work done in this class must be done on an individual basis.
- Credit will be given only for work done entirely on an individual basis.
- Do not make any assumptions as to who can provide help on a programming assignment.
- All submitted work is archived. All submitted programs will be subjected to automated cheating analysis via the MOSS system.

Evidence indicating the violation of the policies stated above will be submitted to the Honor Court.

It is much easier to explain a poor grade to parents or a potential employer than to explain an Honor Court conviction.

If you have questions or are unclear about what constitutes academic misconduct on an assignment, please speak with me.

I take the Honor Code very seriously in this course. The normal sanction I will recommend for a violation of the Honor Code is an **F*** sanction as your final course grade. The F represents failure in the course. The “*” is intended to identify a student who has failed to uphold the values of academic integrity at Virginia Tech.

A student who receives a sanction of **F*** as their final course grade shall have it documented on their transcript with the notation “FAILURE DUE TO ACADEMIC HONOR CODE VIOLATION.”

You would be required to complete an education program administered by the Honor System in order to have the “*” and notation “FAILURE DUE TO ACADEMIC HONOR CODE VIOLATION” removed from your transcript.

The “F” however would be permanently on your transcript.

In recent terms we have observed a new behavior regarding cheating.

Some of the projects we use each semester may have been used in previous offerings of the course.

A small number of students have submitted solutions that were based (in whole or in part) on solutions submitted by other students in previous offerings. Be advised:

- That is cheating.
- When we do the cheating analysis for a project, we include submissions from previous offerings as well as the current term. (We save everything.)
- If we detect this form of cheating, the students from the previous terms will also be charged with violating the Honor Code.
- The University does make provisions for cases in which a charged student has already graduated. You do not want to discover how this works, so safeguard your code for the long term!

At some point you will very likely want to make samples of the code you write available to potential employers. There is absolutely nothing wrong with that!

But if you do that, you need to be sure you're not making it easy for other students to access your code.

GitHub is an interesting service, but you need to use the protection options that the free student accounts offer.

And there are alternatives...

Students are encouraged to address any special needs or special accommodations with me during the first two weeks of the semester, or as soon as you become aware of your needs.

Those seeking accommodations based on disabilities must obtain a Faculty Letter from the Services for Students with Disabilities office (540-231-0858) located in Lavery Hall, Suite 310) <http://www.ssd.vt.edu/>.

If you need adaptations or accommodations because of a disability (learning disability, attention deficit disorder, psychological, or physical), if you have emergency medical information to share with the instructor, or if you need special arrangements in case the building must be evacuated, please meet with the instructor as soon as possible.