CS 3214 Syllabus

Overview

CS 3214 provides an introduction to computer systems as they are relevant to application programmers today, with an emphasis on operating systems principles. Topics covered include: operating systems design and architectures; processes, threads, synchronization techniques, deadlock; CPU scheduling; system call interfaces, system level I/O and file management; shell programming; separate compilation, loading and linking; inter-process communication (IPC); virtual and physical memory management and garbage collection; network protocols and programming; virtualization and cloud computing. The topics will be accompanied by a series of exercises and programming projects that will give you hands-on experience in interacting with systems at different levels. The projects will be done in groups of 2, which gives you the added benefit of learning how to work in a small team.

Online Format

During the Fall 2022 semester, CS 3214 will be taught as an in-person class augmented with various modes of online communication. This consists of in-class meetings/discussions/lectures, pre-recorded lectures, mini-lectures, and demos, book reading assignments, slide sets, use of Discourse, Discord, and Zoom for in-person and online office hours.

Staff Information and Meeting Times

Instructors:	Godmar Back Huaicheng Li Ahmad Yazdani
Office hours:	Available on class website
Class website:	https://courses.cs.vt.edu/cs3214/fall2022
TAs:	Available on class website TA office hours will be held either in person or online.
Email:	To contact the teaching staff, use cs3214-staff@cs.vt.edu or use the discourse forum.
Forum:	We will be using Discourse this semester at https://cs3214.cs.vt.edu. Please join via the Google sign-up and use your @vt.edu email to create an account. Announcements will be posted on the Discourse site.

Class Meeting Times:

Dr. Back’s section: TR 12:30pm-1:45pm WHIT 300
Dr. Li’s section: TR 2:00pm-3:15pm Surge 104C
Mr Yazdani’s section: TR 5:00pm-6:15pm Surge 104D

Attendance at class meetings is not required, but strongly encouraged. Class meetings are not recorded.

Prerequisites

The formal prerequisites for this class consist of a C or better in CS 2506 Introduction to Computer Organization II and CS 2114 Software Design and Data Structures. Prerequisite checking is done per the departmental procedures.

Note that CS 2505 is indirectly a prerequisite for CS 3214. If you received transfer credit for CS 2506, but have not taken CS 2505, you are strongly advised to complete CS 2505 before attempting CS 3214.

Objectives

Upon completion of the course, students should be able to:

1. Explain the major components of an operating system and their respective functionality and interaction, including runtime libraries and systems programs
2. Use an operating system’s command line interface to run shell scripts and perform productivity tasks
3. Describe the theories underlying CPU scheduling and their impact on applications
4. Describe proper synchronization techniques, deadlock detection and avoidance
5. Describe the principles underlying virtual memory and their impact on applications
6. Design, implement, test, and debug:
   a. programs that interact directly with an operating system using its system call application programming interface (API)
   b. multi-process applications that use inter-process communication
   c. multi-threaded applications
   d custom user-level memory allocators
   e. simple network applications
7. Describe the principles underlying virtualization

Tentative Schedule

> *This schedule is tentative. The times and dates posted on the website are authoritative. However, please consider this a heads-up. Unless an announcement is made that these times will be changed, those will be the deadlines.*

Week	**Date**	Due Date	Test Date	Topic Outline
1	8/22			Intro, Getting Started
2	8/29	9/2 ex0		Process Management, Shell, Job Control, Unix Signals, Pipes, File Descriptors, Standard I/O
3	9/5			
4	9/12	9/13 ex1		
5	9/19			
6	9/26	9/28 p1		Compilation, Loading and Linking
7	10/3	10/4 ex2		Multithreading, Concurrency, Synchronization
8	10/10	10/11 ex3		
9	10/17			
10	10/24	10/24 p2 10/28 badge1	10/27 Midterm	Memory Management: automatic & explicit; Virtual Memory
11	10/31			
12	11/7	11/9 ex4		Networking and Socket Programming
13	11/14	11/14 p3 11/18 No-new-topics[1]		
	11/21	Thanksgiving		
14	11/28			Virtualization, Containers, Cloud Computing
15	12/5	12/7 p4,	Final Exam	

[1] After this date, no new topics may be started on Discourse for the purposes of earning a higher-level contribution badge. You may continue to contribute to ongoing topics/threads and you may continue to ask nuts-and-bolts questions.

		ex5, badge2	see Hokiespa	

Textbook

The required textbook is:

Computer Systems: A Programmer's Perspective, Third Edition
(CS:APP3e) Randal E. Bryant and David R. O'Hallaron
Prentice Hall, 2016, ISBN 978-0134092669.

Course Format

The course work consists of a mix of lectures, tests, exercises, and programming projects.

Tests/Final Exams**:** There will be 1 midterm and 1 final exam. Each exam will cover material from the lectures and textbook. The tests may also include questions related to the programming projects and exercises. The final exam will be comprehensive and include material from the lectures, textbook, and programming projects. Midterm and the final exam will be closed book, closed Internet, but a sheet of prepared notes will be allowed. The midterm will be during a class period whereas the final exam will take place at the scheduled time as per HokieSpa.

Exercises**:** Exercises are done individually. They consist of smaller tasks that reinforce necessary skills. Exercises may run in parallel with projects, but generally have a shorter deadline.

Projects**:** There will be 4 medium-size projects. Projects must be done in groups of 2 students. Teaming up with a partner **is not optional.**

There are 2 choices for forming a group. First, you may self-select a partner, then register your group with us. Second, you may choose to be assigned a partner.

You may switch partners, but only between projects. You may work with at most one partner on a given project.

Students must contribute equally to the project within a team. It is not acceptable for students to either not contribute to the project or not to let the other group member contribute equally to the project.

For each project, we will set a deadline by which you must notify us if your assigned or selected group member is not contributing. If you have informed us by the deadline, and your assigned group member ends up not contributing for whatever

reasons, we will provide a cushion for this project, either in the form of a 10% bonus or 3 extra late days at your choice.
We also do not want to encourage students to ignore a lack of contribution by their partner and to attempt to “solo” the project. If that happens, we reserve the right to deduct 10% from the score.

	Self-selected groups must register by	Assignment by	Report problems with partner by	Due
p1	9/5	9/7	9/12	9/28
p2	9/30	10/2	10/4	10/24
p3	10/26	10/28	10/30	11/14
p4	11/16	11/18	11/27	12/7

Projects and exercises will be submitted electronically and grades will be posted electronically. Instructions are available on the course website.

Participation Requirement: To encourage participation and engagement, we set aside a small portion of your grade for participation. To earn this credit, you are expected to contribute to the class on the Discourse forum. A contribution can be asking a question or providing an answer to the class on the Discourse forum. A question or answer should not have been asked or given before.

We exclude questions related to course logistics, such as “when is this assignment due?” “Are we meeting today?” “Is rlogin down?” or “Where can I find the handout for X, Y, Z” - your questions should be related to the topic of the class. (Logistics questions should be sent via email or via private message on Discourse.)

To recognize participation, we will give badges for your contributions using the Discourse badge system. There are 2 kinds of badges:

- A “nuts-and-bolts” contribution or question typically relates to the understanding and/or details of the lectures, projects, and exercises. Examples include clarification questions, questions about APIs, or questions related to basic Unix skills. Examples may include:
  - What does the tilde (~) character do in bash?
  - What does this compiler error mean?
  - How do I achieve buffering?

- A “higher level” contribution badge will be given for contributions that show more effort in terms of preparation and/or insight, either related to the class material or to general, systems-related topics. Examples would include:
  - How can the constructs taught for multi-threaded programming be applied to the parallel machine-learning applications I’m working on? My thoughts on this topic are …
  - How do cloud providers guarantee the availability of the user data they store? Based on my research (... included some basic facts you’ve learned doing initial research.)
- We generally give only one badge per participation in a Discourse topic, unless you make multiple, independent contributions.
-
- We discourage questions that are vague and that show no or too little effort on the part of the questioner to explore the topic themselves first, such as: “What are the trade-offs of using C vs other languages?” If you wish to discuss a topic like this one (and earn participation credit for it), you should start your question by including your own research and your own arguments, then form a position and present it. This may jump start a discussion.
- Read what others have posted before you start a new topic. We may close topics (and not award badges) for duplicate threads.
- Your contribution to an ongoing thread should be (at least partially) novel and not repeat what was already said. Moderators may close topics if necessary.
- To spread and manage the contributions over the semester, we will provide 3 deadlines. We expect everyone to have earned a first badge by the badge1 deadline, and a second badge by the badge2 deadline. In addition, we will set a third deadline after which no new “higher-level” contributions topics may be started. This is intended to prevent students from attempting to meet their contribution requirements by starting discussion topics at the end of the semester when there is no more time to actually discuss them. You may contribute to existing topics until the badge2 deadline.
- To earn full credit, you must earn at least 2 badges by their respective deadlines. One of your badges should be a higher level contribution badge.
- Instructors will reply to questions in a timely manner to make sure that questions related to exercises and projects are answered quickly, but may intentionally delay responses to other questions to give students time to earn participation points.

Late Policy

Deadlines are necessary for everyone to stay on schedule and to facilitate grading. We generally try to avoid granting individual late days and instead use built-in late days every student can individually budget.

Each student will have a **budget of 4 late days that can be used to submit projects late without penalty**. You decide when you want to use your late days –

there is no need to contact the instructor or GTA beforehand. Late days are granted in whole integer multiples of days: if your assignment is 5 minutes late, you may have used up an entire late day, although there may be a short grace period of unspecified length.

In addition, you **have an extra, individual budget of 4 late days you may use on exercises**. Since projects are group endeavors and exercises are done individually, you cannot substitute one type of late day for the other.

For projects, you will work with a partner. If you are working with a partner, late submissions will count against the (project) budgets of both partners, so make sure that all of you have enough late days left or the partner with an insufficient number of days risks not receiving credit. **Submissions received after all late days are exhausted will not be graded and receive zero credit.**

Accommodations beyond automatic late days (aka "extra late days"):
The automatic late day system is designed so that we do not have to respond to requests for additional late days. However, we will provide additional late days in 2 situations:

DoS/University accommodations: If you have family or other emergencies that prevent you from submitting assignments on time, please contact the Dean of Students Office (https://www.dos.vt.edu). They will make a determination as to what accommodations should be given, and inform the instructors of the classes in which you are enrolled of their decision. Our policy is to provide you with as many additional late days as the note from DoS advises. For reasons of consistency and fairness, we will not make any determinations about emergency accommodations; we will defer all such decisions to the DoS. If you have learning or other disabilities, please also see the section Students With Disabilities below.

Sickness policy: If you cannot complete an assignment due to medical reasons, including illness, required isolation or quarantine, **you must tell the teaching staff how many days you need to catch up on any work you were not able to do because of sickness.** The deadline for the assignment in question (and only this assignment) will then be moved by this many days without counting against the late days described above. For group projects, sick days will be granted to the group as a whole for the project in question. No Doctor's note is required or expected, but the honor code and the university policy on class attendance apply. If the overall number of requested sick days throughout the semester is more than 5, we will require an absence verification from the Dean of Students.

Incomplete Policy/Academic Relief

In most cases, students who missed too much work will be asked to withdraw or drop and then retake the course. In exceptional circumstances, the College of

Engineering may grant academic relief. As per updated guidance from the College of Engineering, such academic relief accommodation may include an Incomplete grade provided that all of the following are true:

- The student has been attending class on a regular basis.
- The student has completed 75% or more of the required coursework.
- With exception for the missing work, the student is passing the course.
- The student has a viable plan and timeline for completing the remaining work.

Grading

The tentative contributions of the different portions to your final grade will be as listed below:

14%	Midterm Exam
16%	Final Exam
40%	Projects
24%	Exercises
4%	Participation
2%	Syllabus Quiz

We may curve the final grades for the course. We guarantee that your letter grade will not be worse than the letter grade you would have obtained under the standard scale. That is, a final average of 90% will guarantee at least an A-, a final average of 80% at least a B-, and so forth.

Beyond that, from past experience, the median final grade for this class typically lies between B and C. In other words, students who consistently perform above the median can expect a B- or better. The population over which this median is taken is all students that started the class, not the subset of students who finished it. Note that we will not know the magnitude of this semester's curve until the final exam has been graded.

Each project has a subset of requirements designated as "minimum requirements." **Autofail rule**: to pass the class, you must meet minimum requirements on 2 or more projects. This will demonstrate a minimum of practical skills we expect our CS graduates to have. Note that the opposite is not true: merely passing the minimum requirements on 2 projects will not guarantee that you pass the class.

Modes of Communication

Website: The class website and the Discourse forum are the primary means of communication from us to you. We will post announcements there, which are binding. We have unfortunately no way to reach you by mass email, but the default notification settings should send you email when a post is made under Announcements.

Forum: You are required to subscribe to the Discourse forum. **You are required to read all posts that appear under the Announcements category.**

When posting in the forum, please follow up when you receive a reply or have solved your problem; do not waste your classmates' time with content-free "never mind" posts. See below for the rules regarding honor code violation and the forum.

Zoom/Discord etiquette: Make sure that you are familiar with how to share your screen using Zoom and Discord. Please make sure that your audio works. The use of a microphone that's part of a headset is preferred to the use of built-in microphones.

Email etiquette: Class-related email should be sent to **cs3214-staff@cs.vt.edu**, with the exception of project-related questions, which should be posted in the forum. The email alias ensures that it reaches both the instructor and TAs, and ensures that it will be archived separately. Generally, you should reserve email for administrative issues such as sick days, or other questions related to policy or your individual situation. For all other questions, the forum is usually the place to go.

Email messages regarding academic issues that are sent from non-VT accounts (e.g., hotmail) will be disregarded for FERPA reasons. Do not send email attachments unless we have asked you to do so. ***Never send project or exercise submissions by email.***

Collaboration Policy and Honor Code

The tenets of the honor code will be strictly enforced in this course, and all assignments shall be subject to the stipulations of the Undergraduate Honor Code. For more information on the Honor Code, please refer to the Undergraduate Honor System Constitution, available online at https://honorsystem.vt.edu/.

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.

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).

If you have questions or are unclear about what constitutes academic misconduct on an assignment, please speak with one of the instructors of record.

We take the Honor Code very seriously in this course. The normal sanction we will recommend for any, even a single, 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.

A second F* will lead to expulsion from the university, as has happened in the recent past, including for students who chose to cheat in CS 3214.

If we suspect that an honor code violation has occurred, we generally file complaints directly with the officers of the Honor System. We are not required to discuss suspected honor code violations with suspected students before filing charges, and typically will not do so.

The following policies regarding collaboration apply in this class.

All submitted work is expected to be the original work of the individual student unless otherwise directed by the instructor. Note the emphasis on "submitted" work – this includes work that is explicitly graded and work that may not be graded, regardless.

Projects are to be the work of the individual student or team as specified. You may discuss general concepts, such as software libraries, Internet resources, or class and text topics, with others outside your team. However, discussion of project solutions, specific code, or detailed report content is an honor code violation.

You are not allowed to post code that is part of your solution on the Discourse forum or on Discord. An exception is a single line if it causes a compile-time or runtime error. Posting debugging output, including backtraces, is ok. You may not post detailed descriptions of your design or solution on the forum or elsewhere. You may not post answers to exercise questions on the forum or elsewhere.

You are required to read-protect your work on shared file space and on the department's git.cs.vt.edu server so students outside of your team will not have access. Failing to do so is an honor code violation.

After the class ends, you are still responsible for not publicly sharing your solutions on github.com or similar sites. Note that it is possible to be the subject of an honor code violation even after you have finished the course.

Borrowing code or hiring someone to perform the work for you is an egregious violation of the honor code. We will use plagiarism detection software such as MOSS to screen out students attempting to do this.

In recent terms, 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 or exercise, 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.
- A charged student will not receive a degree until the charges are resolved.
- Despite the clearly spelled out warnings in this syllabus, in certain recent semesters, we were forced to file multiple cases with the honor system. If you heard through the grapevine that some students got away with cheating, consider that you may not have heard from those students who didn't.
- Honor code panels and the Office of Academic Integrity (which performs a subsequent review) typically uphold the instructor's recommendation of an F*.
- If you already have an honor code violation on your record from a previous class, be aware that a second violation (even if otherwise minor) will lead to expulsion (not suspension) from the university, except in rare, extenuating circumstances. The keyword here is "rare" - extenuating circumstances that are not rare will typically not result in a lowering of sanctions.
- Note that the honor system does not distinguish in any way between giving and receiving unauthorized aid - both acts are considered cheating and punished equally. Consider this when a friend asks you to share their solution from the current or a past semester.
- Some students have in the past tried to engage in code obfuscation exercises: renaming variables and functions, rearranging logical expressions or if/else statements or code structure otherwise to try to hide the fact that their code was a derivative of someone else's submission. Be advised that such attempts did not lead to a change in outcome when those students were caught and referred to the honor code system. Honor

code panels have so far always trusted the judgment of the instructor whether unauthorized collaboration has occurred.

- If you think you could make use of code you found online, keep in mind that another student may come up with the same idea. If both of you do, you will both be referred to the honor code system, and we will not need to identify who copied what from whom and when as the sanction will be the same.

Not having read the honor code and its stipulations is no excuse for violating it.
If you have any doubt about what is and is not allowed, it is your obligation to ask the instructor beforehand.

Policy for students retaking the class

Students retaking the class can use code that was created previously only as a reference for themselves, but they are not allowed to share it with their new group members, be it electronically or otherwise. Any project submission for this semester has to be created from scratch by all group members.
We will enforce this policy: you can expect an honor code case to be filed if we detect similarities between your submission and a submission made in previous semesters, especially if the submissions involved different group members.

Students with Disabilities

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 should 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 accommodations because of a disability (learning disability, attention deficit disorder, psychological, or physical), please contact the instructor as soon as possible.

Mandatory Reporting Statement: University Policies 1025/1026

University Policies 1025 and 1026 mandate employees with instructional responsibilities to report all suspected instances of discrimination, harassment, sexual harassment and/or sex/gender-based violence to the Office for Equity and Accessibility (OEA). The University also has identified certain offices as confidential resources. The University's confidential resources include: the University Ombuds Office, The Graduate School Office of the Ombudsperson, Schiffert Health Center, Cook Counseling Center, and the Virginia Tech Women's Center. Additionally, the

University offers supportive measures to survivors of sex and gender based harassment and violence, regardless of whether the survivors file a Formal Complaint. More information is available from the OEA.