

**COMP 421/ELEC 421, COMP 521/ELEC 552**  
**Operating Systems and Concurrent Programming**  
**Course Syllabus**

## Course Description

This course provides a basic understanding of the software that manages a computer's hardware resources to provide a powerful abstract interface on which user programs execute. The course focuses on the control and utilization of processor, memory, storage, and network resources. The concepts in this course include operating system structure, process management and scheduling, interprocess communication, synchronization of concurrent processes, deadlock, main and secondary storage management, virtual memory, file systems, protection and security, and some introduction to networking.

The course content is structured into two parts, a *principles* part and a *projects* part. The lectures and projects have been sequenced so that by the time you are working on a project, we have covered the concepts involved in that project in the lectures. Your implementation work in the projects will help make those concepts "real" and anchor these ideas in your understanding.

There are two versions of this course: COMP 421/ELEC 421, intended for *undergraduate students*, and COMP 521/ELEC 552, intended for *graduate students*. The lectures for both versions meet together, and the projects and exams for both are the same. However, for students taking the graduate version of the course, COMP 521/ELEC 552, there will also be a written report required as a part of each project.

## Class Meetings

Tuesday and Thursday, 2:30–3:45, Herring Hall 100.

## Instructor

Dave Johnson, [dbj@cs.rice.edu](mailto:dbj@cs.rice.edu), DCH 3007, 713-348-3063.

Office hours: Tuesdays, 3:45–4:45 PM, HRG 100 and/or DCH 3007. Specifically, class ends at 3:45, and in past years, there has not been another class scheduled in this classroom after our class. I will remain in the classroom every Tuesday as long as students remain with questions or until we get thrown out of the room for some other event, after which I will be in my office for the remainder of my office hours that day.

## Teaching Assistants

TAs for the class will be announced on the course website shortly.

## Prerequisites

Students taking this course should have already taken the Rice courses COMP 215 and COMP 321, or equivalent. In particular, you must be familiar with data structures and basic computer architecture concepts. You must also be proficient in programming in the C programming language on UNIX/Linux systems.

Most importantly, all programming projects in this course must be done in C, and the projects will require a significant amount of programming. If you need to review your programming skills, pay special attention to studying data structures with pointers. Operating systems, in general, use *more* pointers (and sometimes in *more confusing ways*) than “normal” types of programs commonly do.

Be sure you understand exactly what a pointer actually *is* and how to make it *point to* something; there is a very important difference between declaring a pointer variable and making that pointer point to something (including allocating what you make it point to). Also be sure you understand the difference between a “static” or “global” variable, an “automatic” variable, and a variable allocated with “malloc” (or “calloc” and related calls).

## Text

The textbook for the course is *Operating System Concepts*, by Silberschatz, Galvin, and Gagne, *Ninth Edition*, published by John Wiley & Sons, 2012.

The textbook should be available in Rice’s bookstore and is available (much less expensively) at various on-line retailers such as Amazon.com. Note that we will cover topics in more depth than in the book, and will also cover some in a slightly different order. There may also be a few additional reading assignments, to be handed out in class during the semester.

## Assignments

The assignments in this course will include three programming projects, plus a midterm exam and a final exam. The midterm exam will be take-home; the final exam will be scheduled by the Registrar.

The exams may cover *any* material covered in the course. This includes the material from the lectures, from the assigned sections of the textbook, from any additional reading assignments handed out, or from the programming projects.

The first programming project must be done *individually*, and the last two projects will be done in *groups of two students*. The projects must be done on the Rice CLEAR system running Linux. For the programming projects done in groups, both partners in a project group must fully understand and participate in designing and programming the solution to the project. For example, there might be a question on an exam that can only be answered well if you have done the projects and really understand them.

Note that for the last two projects, done in groups of two students, *both students in a group are jointly responsible for what the group submits for their project*. Only one of the two students in a group actually does the project submission, but both partners should agree on what is to be submitted. Also, after a project submission, both partners should receive email (also sent automatically to the instructor) detailing the submission and the files that were included in it. Please carefully check this email when you receive it to make sure that the right files were submitted. In any case, if you have a concern about what your partner has submitted for your group, try to work it out with your partner (and if needed, you can redo the submission with a different version of the project files); if you still have a concern about a submission by your group, contact the instructor.

In addition, as noted above, for each of the three projects, students taking the graduate version of the course will also be required to submit a written report on the project. This report must be done *individually*, even for the second and third projects, where the other aspects of the project are done in groups of two students; within a project group, each student taking COMP 521 or ELEC 552 must write their own report individually, *not* jointly with your group partner on that project.

## Grading

Your final grade for the course will be computed based on the following tentative weights for the individual assignments:

10%	First programming project
20%	Second programming project
20%	Third programming project
25%	Midterm exam
25%	Final exam

In addition, your weighted project average and weighted exam average must each be a passing grade in order to pass the course.

Any requests for a regrade on an assignment *must* be submitted within one week (7 days) after grades for that assignment are available (either when the graded assignments are returned to the class, when the grades are emailed to you, or when the graded assignment is made available for you to pick up). Requests for regrades made after this one-week cutoff will *not* be entertained unless under extenuating circumstances.

For any regrade request on a *project*, you should contact the teaching assistant who graded your project. After working with that TA on your grading concerns, if you are still not happy, see the instructor. For a regrade request on an *exam*, you must submit to the instructor your *entire original grade exam* and a *written* statement of which questions(s) on the exam you want to be regraded and, for each of these questions you are requesting, a clear statement of specifically why you believe the original grade is incorrect.

## Course Schedule

A *tentative* schedule for major events in the course is available on the course website at:

<https://www.clear.rice.edu/comp421/schedule.html>

## Work Load

This course requires a substantial amount of work, particularly in the projects. My best advice to you is to start each assignment *early*; don't wait until the last few days to try to do all the work. You will need to start on the programming projects early in order to make good use of your time during the assignment; for example, it may take some time while working on the project to pause and understand why something may not be working correctly before proceeding with the next part of the project. I also recommend you work on each project *incrementally*, implementing and testing each part of the project as you go, before proceeding with implementing and testing the next part of the project. The projects can be enriching if you stay on top of them; they can be impossible if you don't.

## Course Web Site

The course web site is located at

<http://www.clear.rice.edu/comp421/>

This course syllabus as well as other up-to-date information about the course will be available via this web site. Online versions of handouts distributed in class will also be available there. Course announcements, such as schedule changes, clarifications to the assignments, and reading assignments will also be posted to the course web site. Please check the web site regularly for announcements.

In addition, we will also be using *Piazza* for class discussion. Piazza is a web-based platform that will allow you to post questions about the course material, including the course projects, and to quickly receive answers from me, from the TAs, and from your fellow classmates. *Please register now for the course on Piazza* by going to

<https://piazza.com/rice/spring2018/comp421>

Throughout the semester, beyond regularly checking the course web site, you should also check Piazza regularly for new information. In addition to finding answers to your own questions, reading other questions and answers on Piazza can help you find additional valuable information including project assistance and clarifications and schedule updates. And if you encounter a new problem or question, you may well find an answer to it by searching on Piazza.

## Honor Code Policy

*The Honor Code is a special privilege and responsibility at Rice University.* As stated in a student editorial published in the January 20, 2016 edition of *The Rice Thresher*: “As incoming students enter Rice, many are surprised by the degree to which the university’s Honor Code extends trust to the student body. . . . The privileges of the Honor Code stem from the idea that Rice’s aim is not just to instill knowledge in its students, but [to] also help them develop moral character. This idea is fundamental to Rice’s identity: Students can and should be held to a high moral character standard.”

Specifically, all assignments in this course are conducted under the Rice Honor Code, a code that you pledged to honor when you matriculated at Rice. You are expected to behave in all aspects of your work in this course according to the Rice Honor Code. When in doubt as to whether a specific behavior is acceptable, ask the instructor for a written clarification. *Suspected Honor Code violations on the projects and/or exams in this course will be researched, documented, and reported in extensive detail to the Rice Honor Council or Rice Graduate Honor Council.* For more information on the Rice Honor System, see <http://honor.rice.edu/> and <http://gradhonor.rice.edu/> . In particular, you should consult the Honor System Handbook at

<http://honor.rice.edu/honor-system-handbook/>

This handbook outlines the University’s expectations for the integrity of your academic work, the procedures for resolving alleged violations of those expectations, and the rights and responsibilities of students and faculty members throughout the process.

In this class, for the midterm exam and final exam, your solutions, of course, must be your own work. During each exam, you may refer to the course textbook, (*Operating System Concepts*, by Silberschatz, Galvin, and Gagne, Ninth Edition, 2012), to any of the handouts from class, and to your *own* notes you made yourself from any source *before* beginning the exam. You may *not* refer to any other sources, such as past exams or things from the web or elsewhere, during the exam.

For the programming assignments in this course, students are encouraged to talk to each other, to the TAs, to the instructor, or to anyone else about the assignment. This assistance, however, must be limited to general discussion of the problem; *each student or project group must produce their own*

*solution to each programming project.* Consulting or copying, in any manner, another student's or project group's solution (even from a previous class or previous year) is prohibited, and submitted solutions may not be copied from any source.

Also, for students taking COMP 521 or ELEC 552, for *all* projects, each student must write their *own* report on the project; for the last two projects, for which the project itself will be done in groups of two students, *each* student in the group must still write their *own* report, *not* working together on your report with your group partner. Submitted reports must not be copied, in whole or in part, from any source, and you must fully cite any references you use in preparing your report.

In addition, for all programming assignments, you may not place source code for your project on any *publicly accessible* repository (such as GitHub), including even after the end of the semester; to do so would be a violation of the Honor Code, as it would give aid to other students on the project. Also, if any such public repositories do exist, you may not refer them (or other such sources) in working on or producing your solution for such project.

*I want to treat you all in this class as responsible adults.* But please be aware that *cheating* on any of the programming projects or exams in this class constitutes a *Rice Honor Code violation*. Submitting a case to the Rice Honor Council requires a lot of work on my part, but it will have a much larger impact on you, your future status as a student at Rice University, and your prospects for a degree from Rice. Please do not make me have to submit any Honor Council cases in this class. This will help both you and me. As I said, I want to treat you all in this class as responsible adults.

## **Policy on Late Work**

Late assignments will not be accepted, except when there are extenuating circumstances. In particular, for the projects, please take project deadlines seriously.

If you are aware of unusual circumstances that preclude a timely submission, submit the assignment early or contact the instructor in advance to arrange for delayed submission if warranted. After-the-fact explanations will not be treated favorably. Sometimes, of course, unusual circumstances may arise unexpectedly, but even in those cases, you should try to alert the instructor (e.g., by email) as early as possible, before the submission deadline.

## **Students with Disabilities**

If you have a documented disability that may affect your academic performance in this class, you should make sure this documentation is on file with Rice's Disability Support Services (located in Allen Center, Room 111, [adarice@rice.edu](mailto:adarice@rice.edu), 713-348-5841) to determine the accommodations you need. You should also meet with me to discuss your accommodation needs.

## **Syllabus Change Policy**

This syllabus is only a guide for the course and is subject to change with advanced notice.