

SYLLABUS

CSC 374 / 407

Computer Systems II

Spring 2024

April 1, 2024

Instructor: Dale Buchholz
Class Time: Monday: 5:45-9:00 pm
Location: Loop Campus – Lewis 1208
Office Hours: Monday: 4:00-5:30 pm or by appointment (Zoom)
Phone: 312-362-6819
Office: CDM 617
E-mail: dbuchhol@depaul.edu
Prerequisite: CSC 373 Systems I
Texts: Bryant & O'Hallaron, *Computer Systems: A Programmer's Perspective, 3rd Edition*, Prentice Hall/Pearson
US Edition, 2016, ISBN 978-0-13-409266-9 (bookstore)
Global Edition, 2016, ISBN 978-1-488-67207-1 (pdf)
Kernighan & Ritchie, *The C Programming Language, 2nd Edition*, Prentice Hall, 1988. ISBN: 978-0131103627

Course Summary:

This course is the second of a two-course sequence covering the concepts underlying all computer systems and how they affect the correctness, performance, and utility of application programming. We will cover the topics of optimizing program performance, linking, processes, exception control flow, system-level I/O and networking, concurrent programming, and web services as a current example of distributed system technology.

Learning Outcomes:

- improve your C programming skills;
- learn techniques for improving code performance;
- understand static and dynamic linking, and how to avoid some common linking errors;
- understand the concept of exception control flow and of a process, and how to use this knowledge when developing applications such as shells and web servers;
- understand the basic concepts of UNIX I/O and how it is used to develop networked applications and distributed systems;
- know how to write concurrent programs and, in particular, multi-threaded programs, and how to develop a multi-threaded web server;
- successful completion will allow you to take advanced, technical CSC and SE courses.

Course Prerequisites:

You must have taken CSC 373 or CSC 406 Systems I or an equivalent course that introduces the Intel Assembly language, system-level programming in C, and computer architecture from an application programmer's perspective. You must also have taken a course that covers basic data structures such as arrays, linked lists, stacks and queues.

- You know how to create, debug, compile and run simple C code on a Linux box, and you use a reasonable programming style (i.e., your code is easy to read and concise).
- You understand how data (e.g., integers, strings, ...) is represented and manipulated at machine-level.
- You can read, understand, and debug Intel Assembly code in GAS (Gnu ASembler) format.
- You understand how programs are represented at machine-level.
- You can manipulate pointers and understand basic implementations of arrays, linked lists, stacks and queues.

Coursework and Grades:

Required coursework components and their contribution to the final grade will be:

- a) 4 Homeworks (50 pts each; total 200 pts)
- b) 4 Labs (50 pts each; total 200 pts)
- c) Mid-Term and Final Exam (100 pts each; total 200 pts)

Further details on each assignment will be distributed in class. Final grades will be calculated as follows: points earned divided by possible points yields a total course percentage between 0% and 100%. Final letter grades will be assigned as:

Percent	Grade	Percent	Grade
93-100	A	77-79	C+
90-92	A-	73-76	C
87-89	B+	70-72	C-
83-86	B	60-69	D
80-82	B-	< 60	F

Late Assignments:

All homework and labs are due on Sunday by 11:59 pm. Late submissions will be penalized as follows: 10% deduction per day late up to 1 week. No credit after 1 week late. D2L will be closed for submission after 1 week. Any penalty may be waived with permission of the instructor. You must notify the instructor via email before the due date in order to get an extension of the due date. You must provide a reason in your email.

Homework assignments, labs and the exams must be completed individually. Class attendance (sync or async) is strongly recommended since lectures may cover topics outside the text. All class sessions will be recorded for asynchronous viewing.

Class Schedule and Text:

Class dates, topics, readings, and assignments are shown in the Class Schedule below.

Class Schedule

CSC 374 / 407 — Spring 2024

(April 1, 2024)

<i>Date</i>	<i>Topics</i>	<i>Readings</i>	<i>Assignments</i>
Week 1 April 1	Course Overview Review of Linux and C	Handout	HW0 due April 7
Week 2 April 8	Memory Hierarchy and Caching	Chapter 6	Lab 1 due April 14
Week 3 April 15	Linking	Chapter 7	HW 1 due April 21
Week 4 April 22	Processes, Exceptions and Signals	Chapter 8	Lab 2 due April 28
Week 5 April 29	Mid-Term out Friday, May 3		HW 2 due May 5
Week 6 May 6	Virtual Memory Part 1	Chapter 9	Mid-Term due May 12
Week 7 May 13	Virtual Memory Part 2	Chapter 9	Lab 3 due May 19
Week 8 May 20	System Level I/O	Chapter 10	HW 3 due May 26
Week 9 May 27	Network Programming	Chapter 11	HW 4 due June 2
Week 10 June 3	Concurrent Programming Final Exam out Friday, May 31	Chapter 12	Lab 4 due June 9
Week 11 June 10	Final (no class)		Due Monday, June 10

Changes to Syllabus: This syllabus is subject to change as necessary during the quarter. If a change occurs, it will be thoroughly addressed during class, posted under Announcements in D2L and sent via email.

Online Course Evaluations

Evaluations are a way for students to provide valuable feedback regarding their instructor and the course. Detailed feedback will enable the instructor to continuously tailor teaching methods and course content to meet the learning goals of the course and the academic needs of the

students. They are a requirement of the course and are key to continue to provide you with the highest quality of teaching. The evaluations are anonymous; the instructor and administration do not track who entered what responses. A program is used to check if the student completed the evaluations, but the evaluation is completely separate from the student's identity. Since 100% participation is our goal, students are sent periodic reminders over three weeks. Students do not receive reminders once they complete the evaluation. Please see <https://resources.depaul.edu/teaching-commons/teaching/Pages/online-teaching-evaluations.aspx> for additional information.

Academic Integrity and Plagiarism

This course will be subject to the university's academic integrity policy. More information can be found at <https://resources.depaul.edu/teaching-commons/teaching/academic-integrity/Pages/default.aspx>.

Academic Policies

All students are required to manage their class schedules each term in accordance with the deadlines for enrolling and withdrawing as indicated in the [University Academic Calendar](#). Information on enrollment, withdrawal, grading and incompletes can be found at: <http://www.cdm.depaul.edu/Current%20Students/Pages/PoliciesandProcedures.aspx>

Incomplete Grades

An incomplete grade is a special, temporary grade that may be assigned by an instructor when unforeseeable circumstances prevent a student from completing course requirements by the end of the term and when otherwise the student had a record of satisfactory progress in the course. All incomplete requests must be approved by the instructor of the course and a CDM Associate Dean. Only exceptions cases will receive such approval. Information about the Incomplete Grades policy can be found at:

<http://www.cdm.depaul.edu/Current%20Students/Pages/Grading-Policies.aspx>

Students with Disabilities

Students seeking disability-related accommodations are required to register with DePaul's Center for Students with Disabilities (CSD) enabling them to access accommodations and support services to assist with their success. There are two office locations:

- Loop Campus (312) 362-8002
- Lincoln Park Campus (773) 325-1677
- Email: csd@depaul.edu

Students who register with the Center for Students with Disabilities are also invited to contact Dr. Gregory Moorhead, Director of the Center, privately to discuss how he may assist in facilitating the accommodations to be used in a course. This is best done early in the term. The conversation will remain confidential to the extent possible. Please see <https://offices.depaul.edu/student-affairs/about/departments/Pages/csd.aspx> for Services and Contact Information.