

CMSC 14300: Systems Programming I

The University of Chicago, Summer 2025

Konstantinos Ameranis

Course Overview

In CMSC 14300, students, building on their experience with applications programming and fundamental data structures, will develop a deeper understanding of a computer's operations when executing a program. In order to make the working of the computer more transparent, we will study the C programming language, with special attention devoted to bit-level programming, pointers, allocation, the stack/heap distinction, memory management, and file input and output. C is the *lingua franca* of computer programmers, and one of the great historic successes in the design of practical programming languages.

We will revisit many basic concepts in programming, such as variables, functions, and types in the context of computer hardware. Furthermore, we will begin an ongoing examination of how memory is organized and structured in a modern machine. We will review basic data structures including strings, arrays, lists, trees, and dictionaries (hash tables) and view them in closer detail than in previous courses.

Students in CMSC 14300 will

- acquire a wide-ranging C programming toolkits, including control structures, functions, enumerations, and **struct** and **union** definitions;
- use static typing to guide program development;
- employ recursion and iteration as complementary methods for repetitious computation;
- understand binary representations of data, and write programs operating at the level of bits and bytes;
- program with pointers, understand pointer arithmetic, and learn how to use indirection operators;
- learn to allocate memory directly and detect and correct memory leaks;
- weigh the advantages and disadvantages of linked and contiguous data structures as alternatives to one another;
- build dictionaries (hash tables) from scratch; *and*
- practice basic and advanced debugging techniques with **lldb** and **valgrind**.

Having learned a second programming language well, and consequently having viewed a variety of common problems from distinct vantage points, students who complete CMSC14300 will begin to see past the superficial characteristics of particular computer programs to appreciate their deeper properties.

Staff

Konstantinos Ameranis (Instructor)

Email kameranis@uchicago.edu

Office JCL 391

Office Hours Mon/Wed 4:00PM-5:45PM, JCL 205

Zoom By appointment

Contacting Us

The communications framework for this course will consist of a combination of online systems: Canvas, Gradescope, and Ed Discussion. It can be a bit confusing to negotiate so many different systems, but to try to keep it simple, we will favor Ed Discussion as the primary point of contact, especially for questions and answers about the course content. We will use Canvas minimally; all information you need will be on or linked in the course website. Finally, Gradescope will be used for grading and managing regrade requests.

If you have questions about the course, and those questions are impersonal — that is, they are about course material or course logistics — we ask that you post those questions publicly on Ed Discussion, rather than contacting any of the staff members directly. This ensures you will receive the fastest, most consistent possible response. Since students usually have common questions, posting public questions is also very efficient for your classmates as well. As yet another advantage, it avoids duplication of work on the part of the staff.

In cases where you have a question that is about your own personal situation and not relevant to the class as a whole, please ask a “private question” on Ed; your classmates will not be able to see any private questions.

It also seems fair to let you know that anonymous Ed Discussion posts are anonymous as far as other students are concerned, but, even in anonymous posts, identities are still visible to the staff.

Lectures

MW 6:00pm-8:30pm (CDT), Ryerson 251

The first lecture is on Monday, June 16; the last is on Monday, August 4.

Schedule of Topics by Week

This is a tentative plan and subject to change. Please use the Schedule page on the course website for details.

Week	Date	Topics
1	6/16/25	Basics: C transition, the terminal, I/O, variables, functions
2	6/23/25	Bits, machine arithmetics, bit-wise operations
3	6/30/25	Pointers, heap allocation
4	7/07/25	Machine structure
5	7/14/25	Arrays, strings
6	7/21/25	Linked lists, sorting
7	7/28/25	Trees, tables
8	8/04/25	Pthreads

Texts

There is no required textbook for this course, but it is useful as a reference to have a copy of *The C Programming Language (Second Edition)*, Kernighan, Ritchie. The textbook is available on campus at the Seminary Co-op Bookstore; you can of course find new and used copies at your favorite online bookstore as well.

Grading

The final grade break-down is shown as follows:

	Percentage
Homework	40%
Quiz	20%
Final Exam	40%

Homework

There will be weekly homework assignments. These will generally be due on Mondays at 11:59pm (CDT). As reflected in the grade-percentage breakdown, homework assignments are not the focus of this course. Homework is intended to for practice and is of paramount importance to your learning. While homework is used in grading it is primarily part of the learning process, not the grading process. Each problem is designed to showcase a desired skill or technique. All assignments in this course are submitted through Gradescope; all regrade requests must be submitted through Gradescope.

Quiz and Exam

There will be a midterm in-class quiz and a final in-class exam.

1. Quiz: **Monday, July 14, 6:00pm-8:30pm** (Tentative).
2. Exam: **Wednesday, August 6, 6:00pm-8:30pm** (Tentative).

Special accommodations for exams should be arranged through SDS. You can contact SDS at any time in advance of an exam; the sooner, the better.

Late Work

Since you submit your work electronically, deadlines are enforced to the minute. You are allowed a total of 4,320 minutes (72 hours) of late submission time in this course. You may use all 4,320 minutes in one assignment, or you may distribute it across multiple ones. You do not need to contact me for an extension if you have not spent all of your late time. No homework will be accepted more than 4,320 minutes late.

A final grade penalty will be applied to submissions that exceed the allowed late time. For every minute past the given 4,320 minutes, 0.003% will be deducted from your final grade, which has a maximum of 100%. This penalty equates 4.32% per additional late day.

If you have an emergency that you believe warrants an exception to this policy, you should contact your college advisor. Once the College is aware of the emergency, you should contact me by email, CC'ing your college advisor, and we will do our best to accommodate your situation. Your well-being and success in class is of the utmost importance to us and these policies are put in place to ensure fairness to others.

Academic Honesty

In this course, as in all your courses, you must adhere to college-wide honesty guidelines as set forth at Academic Integrity & Student Conduct. The college's rules have the final say in all cases. Our own paraphrase is as follows:

1. Never copy work from any other source and submit it as your own;
2. Never allow your work to be copied or seen by another student;
3. Never submit work identical to another student's;
4. Never look at someone's working solution in order to solve a problem;
5. Document all collaboration; *and*
6. Cite your sources.

We are serious about enforcing academic honesty. If you break any of these rules, you will face tough consequences. Specifically, any student who is determined to have participated in academic dishonesty will not be allowed to withdraw and will receive a course grade no higher than a C.

Please note that sharing your work publicly (such as posting it to the web) breaks the second rule. With respect to the third rule, you may discuss the general strategy of how to solve a particular problem with another student, in which case you must document it, but you may not share your work directly, and when it comes time to sit down and start typing, you must do the work by yourself. If you ever have any questions or concerns about honesty issues, raise them with your instructor early.

Large Language Models

No code or solutions may be copied or paraphrased from an LLM. This course aims at helping you develop basic skills in Systems Programming. By the end you are expected to have an intuitive

understanding, which is hampered by receiving the solution from a third party. We do encourage you to use the textbook, internet and LLMs to further understanding and explore topics, but **any submitted work must be your own.**

Accessibility Statement

The University of Chicago is committed to ensuring equitable access to our academic programs and services. Students with disabilities who have been approved for the use of academic accommodations by Student Disability Services (SDS) and need a reasonable accommodation(s) to participate fully in this course should follow the procedures established by SDS for using accommodations. Timely notifications are required in order to ensure that your accommodations can be implemented. Please contact Víctor Almaraz Argueta, the Course Coordinator, to discuss your access needs in this class after you have completed the SDS procedures for requesting accommodations.

- SDS Phone: (773) 702-6000
- SDS Email: disabilities@uchicago.edu
- Course Coordinator: jesusaa@uchicago.edu