

Besides the policies in this syllabus, various University policies may apply to students during the semester. Policies that may be relevant appear in the Undergraduate Catalog, at www.umd.edu/catalog.

1 Prerequisites and description

Prerequisite: C– or better in both CMSC 132 and MATH 141

Corequisite: CMSC 250 (or equivalent)

Credits: 4

The goal of the course is to convey the fundamental concepts that enable programs to execute on real hardware. These include how the operating system virtualizes the hardware to provide basic services and abstractions to allow a user program to effectively use the available hardware resources. The course also addresses how different programming constructs and idioms work. The basic abstraction of a program running as one or more threads of control in a single flat address space (a Unix *process*) is the key to the course. Emphasizing that abstraction as the underlying model for understanding how a program works, from both the user program and hardware perspective (with the operating system in between), runs as a theme through the course.

2 Course topics (SUBJECT TO CHANGE)

The following list of topics may vary according to the pace of lecture, so their order and duration are approximate.

Topic	Approx. time
Course introduction	1/2 lecture
The UNIX memory model, and moving from Java to C	4 1/2 lectures
Pointers, strings, and dynamic data structures in C	4 lectures
Make, the preprocessor I/O, and standard libraries	3 1/2 lectures
I/O and standard libraries	1 lecture
Assembly language	4 lectures
Process control and systems programming	3 lectures

Topic	Approx. time
Multithreaded programming with pthreads	2 lectures
Program measurement and optimization (1/2 week)	1 lecture
Libraries and linking	1 lecture
Data representation	1 lecture
Testing	1 lecture
Implementation of dynamic memory management	1/2 lecture

3 Course materials and textbooks

Course materials will be made available via the University's ELMS site www.elms.umd.edu. All registered students automatically have access to ELMS for this course. Essential announcements may be made via ELMS.

The required textbook is *Pointers on C*, Reek, Addison–Wesley, 1998; ISBN 0–673–99986–6. Note that this text is listed as required. At times there may be graded coursework on material that has not been covered in lecture, that students were assigned to read about in the textbook. If you don't purchase the textbook yourself you need to make arrangements to have access to it some way, for example being able to borrow a friend's.

Some course material (mostly in the second half of the semester) will be drawn from *Computer Systems: A Programmer's Perspective, 2nd edition*, Bryant and O'Hallaron, Prentice Hall, 2010, ISBN 10: 0–13–610804–0. This text is neither required nor recommended. You may purchase it if you want, but we will place a few copies of it on reserve in the Engineering & Physical Science (Math) Library, where students can read it for a couple of hours at a time if desired.

4 The instructional staff, email, and office hours

4.1 Instructors

	Larry Herman
sections:	010X, 020X, and 030X
office:	3211 A.V. Williams
phone:	(301) 405–2762

	Dr. Anwar Mamat
sections:	040X
office:	3259 A.V. Williams
phone:	(301) 405–2696

4.2 Teaching assistants

name	duties
Joan Zhang	teaching, 0101
Majeed Kazemitabaar	teaching, 0102 & 0103
Matthew Evanusa	teaching, 0104 & 0303
Adi Acharya	teaching, 0201 & 0301
Mahmoud Sayed	teaching, 0202 & 0203
Somay Jain	teaching, 0204 & 0403
Jesse Moll	teaching, 0302
Ahmed Taha	teaching, 0401 & 0402

name	duties
Angela Kuriacose	grading
Chijioke Okonkwo	grading
Darshan Shah	grading
Hao Li	grading
Jackie Jiang	grading
John Mangino	grading
Sirius Chen	grading
Phil Feelemyer	grading
Ujjwal Goel	grading

4.3 Email

In a class of 400 students the instructional staff is not able to explain most course material via email; class discussion or personal communication are more appropriate. Administrative issues as well as course material should be discussed in person whenever possible (office hours, and before and after class are good times). Send us email in case of urgent or emergency matters only. It is also not feasible in a large course, to provide information or assistance regarding programming assignments via email, so ask any questions about coding assignments in person during office hours.

The instructional staff will only read and reply to emails sent via the ELMS message system (click on *Inbox*). Due to the size of the course, we will typically read and reply to messages a few times a week. If you have a question or issue that takes an exchange of several messages to answer or resolve, it could take over a week to handle it via email. You will get a quicker response by discussing issues in person, before or after class, or during office hours.

It's understood that students may on occasion be unable to attend class for various reasons, but email (and office hours) are **not intended as a replacement for class attendance**. If you miss class for any reason, even with an excused absence, the course size makes it infeasible for the instructional staff to explain via email what you missed. Instead, you would be responsible for finding out what was missed and getting notes on your own from a classmate.

4.4 Office hours

Office hours will be provided in a separate handout shortly, and the course office hours will begin when it is provided.

Students who **regularly attend class** will be eligible to receive assistance during office hours.

While assistance is available for programming assignments from the TAs during office hours, you are ultimately responsible for developing and debugging them on your own, as they are your coursework that you're receiving a grade for. It is not the instructional staff's responsibility to get your assignments to work. If you come to office hours for help with program debugging, expect the TAs to guide you in the right direction, then it would be up to you to continue working on the problem on your own.

5 Computing environment and project submission system

Unlike earlier courses, the program development environment will not be the Eclipse IDE, but rather command-line use of Linux. Programming will be done on the Division of Information Technology's Grace Cluster, which can be accessed at grace.umd.edu. Students will use their own TerpConnect accounts to access the Grace cluster and do coursework, so those who need to should activate their TerpConnect immediately at www.it.umd.edu/new.

Programming assignments will be submitted to the same CMSC project submission and testing server used in the preceding courses, available at <https://submit.cs.umd.edu>. However, a different mechanism than before will be used to turn programs in to the server. Details will be provided with the first programming assignment.

6 Coursework and exam dates

6.1 Coursework

Coursework will count toward the final grade according to the following percentages:

Midterms:	two midterms	26%	(equally weighted)
Final:	will be comprehensive	26%	
Programming projects	six or seven expected coding assignments	26%	
Quizzes	in discussion section or lecture	8%	(equally weighted)
Practice exercises	short coding exercises either done in discussion section, or handed out in discussion and due the next day	8%	(equally weighted)
In-class (discussion or lecture) worksheets	these may be individual or group work	6%	

Quizzes will be given in discussion section or lecture and will cover discussion and lecture material. They will be announced in advance, usually in the prior discussion section.

Practice (programming) exercises will usually provide practice with concepts needed for later projects. Typically they will be handed out in discussion section and not announced in advance.

Unannounced graded worksheets may be done in lecture or discussion section; these may be group exercises.

Ungraded homework problems and solutions will periodically be provided, to allow you to test your knowledge of the material. It's suggested you study the relevant material and work on these problems when topics are first covered, to ensure you are prepared for later graded coursework using it. If you have questions about the problems, or have trouble solving them, ask during office hours (or discussion section, if time permits).

If you experience difficulty during the semester keeping up with the academic demands of your courses, you may consider contacting the Learning Assistance Service in 2201 Shoemaker Building at (301) 314-7693. Their educational counselors can help with time management issues, reading, note-taking, and exam preparation skills.

6.2 Exam dates

Midterm exams will be held during lecture. The dates below will be confirmed in advance in class, and might vary depending on lecture progress and other factors (in particular, if the University has any unexpected closures).

The final exam will be rescheduled **only** for students having another final at exactly the same time, or for students with more than three final exams scheduled on the same day. (The only courses that students who are enrolled in CMSC 216 should be able to take that have finals at the same time as its final are BIOM 301, EDMS 451, and ENMA 300/ENME 382.) If either of these situations applies to you, you must inform your instructor **at least two weeks in advance** of the final exam time for allowances to be made. Also please inform your instructor immediately if you have a conflict with a scheduled midterm date, or any other important date as the semester progresses.

Exam #1: Thursday, October 15
Exam #2: Tuesday, November 17
Final exam: Monday, December 14, 4:00-6:00

7 Class and attendance, and grading

7.1 Class and attendance

Class locations and times are available on Testudo at <https://ntst.umd.edu/soc>. Students are responsible for all academic and administrative material discussed in lecture and discussion section, whether they were in class to hear it or not. Other than cases of excused absences (see Section 9.1 below), or University cancellations, students are expected to attend all lectures and discussion sections.

After the two-week drop/add period is over attendance will be taken in Larry Herman's lectures. Laptop computers, tablets, and cell phones may not be used during Larry Herman's lectures.

7.2 Grading and minimum project requirements

A student who feels that something was graded incorrectly on an exam may provide a written (not emailed) explanation within a week of when the exam is returned and solutions are provided. Keep in mind that the exam may be regraded in its entirety, and if upon consideration of the explanation it is determined that the answer deserved fewer points than given in the original grading, the grade will be adjusted accordingly. In other words, if a request is made for specious reasons the grade may be lowered, so it is in your interest to check your answers carefully, and make sure that something is really more correct than how it was graded, before asking for the grading to be reconsidered.

Final course grades will be curved as necessary, based on each student's total numeric score for all coursework at the end of the semester. In other words, individual assignments or exams will not be curved; just the final course grades.

All projects will be graded out of 100 points, but depending upon their relative difficulty, which isn't known in advance, they may not be weighted equally. Therefore the relative weights of the projects will be given near the end of the semester. A handout with the project submission and grading policies will be provided when Project #1 is assigned.

This is a required programming course that is part of the CMSC and ENCE curriculum, which teaches language details and concepts essential for later CMSC courses. As a result, the ability to submit reasonably successful versions of the projects is necessary. Therefore a student **will not** be able to pass the course with a grade of C– or higher unless each of his or her projects meets the minimum requirements to be explained in the project submission and grading policies. Even if the final deadline for a project has passed, if none of your submissions satisfies these requirements you **must still submit a version that does, by the end of the semester, in order to be eligible to pass the course**, although you will not receive any additional credit for it. Discuss any questions with your instructor in person.

8 Course evaluations and feedback

Course evaluations are important, and the department and instructors take student feedback seriously. Complete your evaluation later in the semester at www.courseevalum.umd.edu. However, rather than waiting until the end of the semester to give feedback, please bring any suggestions or concerns to our attention during the course, in person whenever possible. Although we cannot guarantee to be able to change anything that is brought up, we welcome hearing any comments or questions, that you may have, and will see if they can be addressed.

9 Absences and accommodations

9.1 What are considered to be excused absences

An excused absence refers to missing class for an approved reason. An excused absence will not affect a student's grade. A student can be excused for an absence from a single lecture or discussion section for medical reasons if:

- They make every attempt to inform their instructor of their illness prior to the class if at all possible; if not, then as early as possible.
- Upon returning to the class, they present their instructor with a self-signed note (on paper, not email) attesting to the date of their illness, which **must** say that the information provided is true and correct, **and** that the student understands that providing false information is punishable under the University's Code of Student Conduct.

(Providing any false information is prohibited under of the Code of Student Conduct– see Part 10 (j) of the policy available at <http://www.president.umd.edu/policies/docs/V-100B.pdf>– and will result in being submitted for disciplinary action.)

Self-documentation of illness may not be used for the major scheduled grading events (the two midterms and the final exam), and may only be used **only once** during the semester. (After using self-documentation for one illness, medical documentation is required for any subsequent excused absence.)

If a student needs to be excused for a prolonged illness (for this course this means missing two or more consecutive class meetings), or if a major scheduled grading event is missed due to illness, written documentation from the Health Center or an outside health care provider **must** be provided. It must include the provider's contact information, dates of treatment, and the time that the student was incapacitated and unable to meet academic responsibilities. (Confidential or diagnostic information about the illness does not need not be included.)

Other University-approved situations, such as religious observance, participation in required university activities, or family or personal emergency, will also be considered excused absences, provided that written documentation of the cause of the absence is provided whenever possible, and the maximum possible advance notice is given. To find out whether a situation may constitute an excused absence, talk to your instructor about it in person.

9.2 How excused absences will be handled

All arrangements for excused absences must be made with the instructors, even if the coursework that was missed, was in discussion section. Any request for an excused absence must be made in writing (not via email).

- In the case of an excused absence for a practice programming exercise or in-class worksheet, extra time will be allowed to submit it.
- An excused absence for an exam will be handled by giving a makeup exam.
- For an excused absence for a quiz, rather than a makeup, the score will be computed as the average of the student's scores for the other quizzes at the end of the semester.

Excused absences do not apply to projects. These will be assigned with sufficient time to be completed by students who have a reasonable understanding of the necessary material and begin promptly, even if an excused absence occurs. In cases of **protracted, extremely major** illness, or severe emergency situations, **short** extensions on projects may be considered, depending upon the circumstances. Contact your instructor as soon as possible in such a situation

9.3 Students with disabilities

An eligible student requesting reasonable academic accommodations due to a disability must provide a letter of accommodation from the Office of Disability Support Services (DSS) to the instructor in the first two weeks of the semester. All arrangements for exam accommodations as a result of disability **must** be made and arranged with the instructor **at least three business days prior to the exam date**, or accommodations cannot be made.

10 Academic integrity

The Campus Senate has adopted a policy asking students to include the following statement on each examination or major assignment in every course: *I pledge on my honor that I have not given or received any unauthorized assistance on this examination* (or assignment). Consequently, you will be requested to include this pledge on exams and projects.

Unless otherwise noted, all graded coursework— including programming coursework— is to be done **individually**, so cooperation or use of unauthorized materials on these is a violation of the University’s Code of Academic Integrity. **Any evidence** of this, or of possible violations of the Honor Code on exams or quizzes, **will be submitted** to the Student Honor Council, which could result in an XF for the course, suspension, or expulsion.

For academic honesty purposes, **programming assignments are to be considered comparable to a take-home exam. That is, any cooperation or exchange of ideas would be prohibited on an exam is also prohibited on a programming assignment**. Note the following:

- In learning the material students are welcome to study together or to receive help from anyone else. It’s OK to discuss with others the languages taught, what was covered in class, **general** debugging techniques, or the requirements of a programming assignment. These things convey no information about how to solve an assignment.
- When it comes to actually designing, writing, or debugging a coding assignment, other than help from the instructional staff, these must solely and entirely be a student’s **own work**. Working with anyone else, or using anyone else’s work in any way (except as noted below) is a violation of the code of academic integrity.

VIOLATIONS OF THE CODE OF ACADEMIC INTEGRITY MAY INCLUDE, BUT ARE NOT LIMITED TO:

1. Failing to do any of the work on a programming assignment by yourself, other than assistance from the instructional staff.
2. Using any ideas or any part of another person’s program, or copying anyone else’s work in any way.
3. Giving any parts or ideas from your program, including test data or test cases, to another student.
4. Transferring any part of a program to or from another student or individual by any means.
5. Putting a program anywhere (for example online) that any other student could access it.

In designing or writing programming assignments, students are free to use information and code provided by the instructional staff, **only** if the source is cited in a comment in the relevant section of the program, only short sections of provided code are used, and the substantial part of a student’s program remains their own individual work.

If you have any question about a particular situation or source, ask an instructor in advance. Should you have difficulty with a programming assignment you should see the teaching assistants in office hours, rather than soliciting help from anyone else in violation of these rules.

Note that after the semester is over, providing copies of your coursework to others who are taking this course in a later semester is still considered an academic integrity violation by the University.

IT IS THE RESPONSIBILITY, UNDER THE HONOR POLICY, OF ANYONE WHO SUSPECTS AN INCIDENT OF ACADEMIC DISHONESTY HAS OCCURRED TO REPORT IT TO THE INSTRUCTOR, OR DIRECTLY TO THE HONOR COUNCIL.

You are encouraged to learn more about academic integrity at the UMCP Student Honor Council’s website, and to read the Code of Academic Integrity for yourself, using the links on the course’s ELMS page. You may read the Office of Information Technology’s policy regarding acceptable use of computer resources at

<http://it.umd.edu/security/Nethics/Policy/aup.html>.

11 Right to change information, and copyright

Although this syllabus is believed to be complete and accurate, errors may be discovered, or situations that arise during the semester could require adjustments later. Consequently, the instructors may need to change information or policies in this syllabus (or in other course materials) if necessary.

Most course materials are copyright Larry Herman and Anwar Mamat (and other CMSC faculty and instructors) © 2015. All rights reserved for these materials. Students are permitted to use course materials for their own personal use only. Materials may not be distributed publicly or privately to any others (excepting other students in the course), in any way or format. The University provides that students who distribute copyrighted material in ways they have not been given permission to are subject to being forwarded to the Office of Student Conduct.