1 Prerequisites and description

This course is 4 credits. Its prerequisites are: a C− or better in CMSC 131 (or a score of 5 on the Java AP Computer Science A exam, or passing the CMSC department’s placement exam), and a C− or better in MATH 140.

CMSC 132 is the second part of a two–semester introduction to object–oriented programming for computer science and computer engineering majors, using Java. It provides an introduction to the use of computers to solve problems using software engineering principles. Students will design, build, test, and debug medium–sized software systems, and learn to use relevant tools. Object–oriented methods are used to create effective and efficient problem solutions. Students will use and implement application programming interfaces (APIs). Some concepts about Java that CMSC 131 did not cover are first explained. Then the major part of the material is an introduction to data structures and algorithms.

Besides the policies in this syllabus, various University policies apply to students. Many policies in the Undergraduate Catalog, which can be accessed by clicking on www.umd.edu/catalog, may be relevant.

2 Class materials and textbook

Course materials will be made available via the University’s ELMS site, at www.elms.umd.edu. All registered students will get access to ELMS for this course automatically. Essential announcements may be made via ELMS; students are responsible for reading all announcements there.


Because this course is one of the largest on campus, the libraries have purchased a copy of the textbook, which is in McKeldin Library. You can check it out for 4 hours at a time by asking for it at the Library Services Desk.

3 The instructional staff

Instructor  
Office 3211 A.V. Williams  
Phone (301) 405–2762

Teaching TAs  
Richard Neal teaching, 0101 & 0102  
Ahmed Taha teaching, 0103 & 0104  
Joan Zhang teaching, 0201  
Jyna Maeng teaching, 0202  
Jesse Moll teaching, 0203 & 0303  
Manaswi Saha teaching, 0204 & 0304  
Steven Schwarz teaching, 0301 & 0302

Grading TAs  
Chijioke Okonkwo grading  
Dan McLaughlin grading  
Darshan Shah grading  
John Mangino grading  
Liang He grading  
Michael Maynord grading  
Neelima Pradhan grading  
Phil Feeleymer grading  
Zehua Zeng grading

3.1 Email contact

In a class of 435 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 email messages sent via ELMS. Due to the size of the course, we will typically read and reply to emails a few times a week. If you have a question or issue that takes, for example, an exchange of three or four emails to answer or resolve, it would take over a week to handle it via email— you will almost always get a quicker answer by discussing issues with us in person.

3.2 Office hours

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

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 on.
for. It is not the instructional staff’s responsibility to get these assignments to work for you. If you come to office hours for help with program debugging, expect the TAs to point you in the right direction, after which it would be up to you to continue working on the problem on your own.

4 Course evaluations

Course evaluations are important, and the department and instructors take student feedback seriously. The system may be accessed near the end of 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.

5 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.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Approx. time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Course introduction</td>
<td>1/2 lecture</td>
</tr>
<tr>
<td>Review of Java language concepts from CMSC 131</td>
<td>4 1/2 lectures</td>
</tr>
<tr>
<td>Some new Java language concepts</td>
<td>2 lectures</td>
</tr>
<tr>
<td>Error handling and exceptions</td>
<td>1 lecture</td>
</tr>
<tr>
<td>Introduction to Java I/O</td>
<td>1 lecture</td>
</tr>
<tr>
<td>Abstract classes</td>
<td>1/2 lecture</td>
</tr>
<tr>
<td>Inner classes</td>
<td>1 1/2 lectures</td>
</tr>
<tr>
<td>Generics</td>
<td>1 1/2 lectures</td>
</tr>
<tr>
<td>Linear data structures</td>
<td>3 1/2 lectures</td>
</tr>
<tr>
<td>Recursion</td>
<td>1 1/2 lectures</td>
</tr>
<tr>
<td>Trees</td>
<td>3 lectures</td>
</tr>
<tr>
<td>Algorithmic complexity (efficiency)</td>
<td>3 1/2 lectures</td>
</tr>
<tr>
<td>Sets and maps</td>
<td>1 lecture</td>
</tr>
<tr>
<td>Hashing</td>
<td>2 lectures</td>
</tr>
<tr>
<td>Heaps and priority queues</td>
<td>1 lecture</td>
</tr>
<tr>
<td>Graphs</td>
<td>3 lectures</td>
</tr>
<tr>
<td>Concurrency and threads</td>
<td>3 1/2 lectures</td>
</tr>
<tr>
<td>Introduction to networking</td>
<td>1 lecture</td>
</tr>
<tr>
<td>Sorting</td>
<td>2 1/2 lectures</td>
</tr>
<tr>
<td>Introduction to software engineering and testing, object–oriented design, algorithm strategies, and design patterns</td>
<td>3 lectures</td>
</tr>
</tbody>
</table>

6 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 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. If you miss class for any reason, even an excused absence, the instructional staff cannot explain via email what you missed; the course size makes this infeasible. Instead, you would be responsible for finding out what was missed and getting notes on your own from a classmate who was present.

After the two–week drop/add period is over attendance will be taken in lecture.
Laptop computers, tablets, and cell phones may not be used during lecture.

7 Coursework and grades

Grades will be recorded on the ELMS system mentioned above, and may be checked for correctness there. Do not make comments on grades in the gradebook; due to the size of the course they will not be read. If you have questions or concerns about any of your grades, ask in person.

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

<table>
<thead>
<tr>
<th>Coursework</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midterm exams: two midterms</td>
<td>27%</td>
</tr>
<tr>
<td>Final: will be comprehensive</td>
<td>27%</td>
</tr>
<tr>
<td>Programming projects: around eight expected coding assignments</td>
<td>26%</td>
</tr>
<tr>
<td>Quizzes: in discussion section or lecture</td>
<td>8%</td>
</tr>
<tr>
<td>Practice exercises: short coding exercises either done in discussion section, or handed out in discussion and due the next day</td>
<td>6%</td>
</tr>
<tr>
<td>In–class (discussion or lecture) worksheets: these may be individual or group work</td>
<td>6%</td>
</tr>
</tbody>
</table>
Unannounced graded worksheets may be done in lecture or discussion section; these may be group exercises. Not all worksheets will be graded.

Besides the graded coursework, ungraded practice problems will be provided as worksheets done during discussion section, and as problems via ELMS. (Solutions to problems on ELMS will also be provided there.) These problems will allow you to test your knowledge of the material and prepare for later graded coursework. If you have questions about these problems, or need help solving them, ask during office hours (or discussion section, if time permits).

Submission and grading policies for programming projects will be provided when the first one is assigned. Projects will be graded out of 100 points, but depending upon their relative difficulty, which can’t be predicted in advance, their weights will not all be the same, and these will be determined at the end of the semester. In order to be able to pass the course a student must submit versions of all projects that satisfy minimum criteria, as the policies will explain.

Programming assignments will be written in Eclipse, and can be done on the machine of your choice, usually your own computer. A handout provided shortly will describe how to install and set up the necessary software. The Division of Information Technology (DIT) has computing labs on campus for those who need to use them; the locations are at www.it.umd.edu/wheretogo.

Practice programming exercises will be short, and able to be done either during discussion section in some cases, or within a day in other cases. These will usually provide practice with concepts needed for later projects.

Final course grades will be curved if necessary, based on each student’s total score for all coursework at the end of the semester. However, whether there is a curve, or it would look like if there is one, cannot be predicted in advance.

A student who feels that something may have been 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.

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.

8 Quiz, midterm exam, and final exam dates

Quizzes will be given in discussion section, and cover discussion and lecture material. They will be announced in a prior class. Practice programming exercises typically will be handed out in discussion section but not announced in advance.

Midterm exams will be given during lecture; tentative dates are below. These will be confirmed later, and may have to be changed depending on lecture progress or other factors. The final exam date and time can only be rescheduled for students having another final at exactly the same time, or having more than three final exams on the same day. (The only courses that students in this one should be able to take that have finals at the same time are BMGT 221 and ENES 221.) If either of these situations applies to you, you must inform the instructor at least two weeks in advance of the final exam time. Also inform the instructor immediately if you have a conflict with a scheduled midterm date.

<table>
<thead>
<tr>
<th>Exam #1</th>
<th>Exam #2</th>
<th>Final exam</th>
</tr>
</thead>
<tbody>
<tr>
<td>Friday, March 4</td>
<td>Friday, April 15</td>
<td>Friday, May 13, 4–6 p.m.</td>
</tr>
</tbody>
</table>

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 the 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 the 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 as defined below, and may only be used only once during the semester (medical documentation is required for a second or subsequent absence).

© 2016 Larry Herman; all rights reserved
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, verify dates of treatment, and indicate the time that the student was incapacitated and unable to meet academic responsibilities, in the provider’s opinion. (Confidential or diagnostic information about the illness does not need not be included.) The major scheduled grading events for this course are the two midterms and the final exam.

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 the instructor about it.

9.2 How excused absences will be handled

All arrangements for excused absences must be made with the instructor, 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 will not affect 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. In such a situation you must contact the instructor as soon as possible.

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 Java language, what was covered in class, general debugging techniques, or the requirements of a programming assignment. These are things that 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 the 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 instructor may need to change information or policies in this syllabus (or in other course materials) if necessary.

All course materials are copyright Larry Herman (and other CMSC faculty and instructors) © 2016. All rights reserved. Students are permitted to use course materials for their own personal use only. Course materials may not be distributed publicly or provided to 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 for are subject to being forwarded to the Office of Student Conduct.