Syllabus

Please read this sheet carefully, and save it for future reference.

Instructor

<table>
<thead>
<tr>
<th>Name</th>
<th>Office</th>
<th>Phone</th>
<th>Email</th>
<th>Office hours</th>
</tr>
</thead>
</table>
| Matthew Hertz, Ph.D. | 352 Davis | 645-4736 | mhertz@buffalo.edu | Tuesday 2:00 PM – 3:30 PM  
                             Friday 8:30 AM – 11:00 AM  
                             OR by appointment |

Teaching Assistants

See course website.

Course Information

Credit hours:
- CSE 116 Introduction to Computer Science for Majors II – 4 credits
- CSE 504 Computer Science for Non-Majors II – 3 credits

Course Website: www.cse.buffalo.edu/~mhertz/courses/cse116

Course Description

This course is a continuation of CSE115/503, in which heavy emphasis is placed on abstract data types (ADT’s) and object-oriented methodology. You will be expected not only to understand abstract data types, but also how to design and implement robust ADT’s using a modern object-oriented programming language (Java). Topics such as encapsulation, polymorphism, interfaces, inheritance and composition will be emphasized. Essential topics to be integrated in this framework include the use of recursion, references, linked structures including linked lists, binary trees, stacks, queues, and other advanced data structures and algorithms, including advanced searching and sorting algorithms. The analysis of algorithm complexity (O-notation) will be introduced.

In the course of working on a team software project you will be introduced to a software development methodology, select software development tools, software documentation and teamwork.

The course website contains a detailed, day-by-day schedule of topics to be covered.

Prerequisites

You must have passed CSE115 (or CSE503) with a minimum grade of C-, or have an equivalent background and receive permission from the instructor.

Textbook and Materials

This course uses TopHat to include interactive questions during the lectures. Students are required to have an active Top Hat account and to bring a laptop, tablet, or cell phone to each lecture. For more information see: www.tophat.com

Readings will be assigned using webpages (links found on the course website) and from the required textbook:

Additional reading material may be assigned during the course, and will be announced in lecture.
Student Learning Outcomes

This course follows recommendations of ACM’s CC2001 curriculum document for a second semester introductory course, covering topics from these knowledge units: PF3 Fundamental data structures, PF4 Recursion, PF5 Event-driven programming, AL1 Basic algorithmic analysis, AL3 Fundamental computing algorithms, PL4 Declarations and types, PL5 Abstraction mechanisms, PL6 Object-oriented programming, HC2 Building a simple graphical user interface, SE1 Software design, SE2 Using APIs, and SE3 Software tools and environments.

This course is required of all computer engineering BS students and addresses the following student outcomes of the BS Computer Engineering program:

(CEN-a) An ability to apply knowledge of mathematics, probability and statistics, computer science and electrical engineering as it applies to the fields of computer software and hardware

(CEN-h) The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

(CEN-k) An ability to use the techniques, skills, and modern hardware and software engineering tools necessary for computer engineering practice.

This course is required of all computer science BS students and addresses the following student outcomes of the BS Computer Science program:

(CS-a) An ability to apply knowledge of computing and mathematics appropriate to the discipline.

(CS-d) An ability to function effectively on teams to accomplish a common goal.

(CS-c) An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.

(CS-i) An ability to use current techniques, skills, and tools necessary for computing practice.

Student Learning Outcomes mapping

In the following, fundamental data structures are taken to include linked and array-based lists, stacks, queues and various kinds of trees, especially binary search trees.

<table>
<thead>
<tr>
<th>Upon successful completion of this course a student will be able to...</th>
<th>CEN-a</th>
<th>CEN-h</th>
<th>CEN-k</th>
<th>CS-a</th>
<th>CS-c</th>
<th>CS-d</th>
<th>CS-i</th>
<th>Assessment Instruments</th>
</tr>
</thead>
<tbody>
<tr>
<td>demonstrate the operation of fundamental data structures</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>HW, Project, Exams</td>
</tr>
<tr>
<td>implement fundamental data structures in Java</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Project</td>
</tr>
<tr>
<td>write unit tests from informal specifications</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Project, Exams</td>
</tr>
<tr>
<td>execute unit tests using the JUnit unit testing framework</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>HW, Project</td>
</tr>
<tr>
<td>employ a versioning tool (git) to manage program files for teamwork</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Project</td>
</tr>
<tr>
<td>use an integrated development environment (Eclipse JDT) to perform typical program editing and navigation tasks</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>HW, Project</td>
</tr>
<tr>
<td>define the O (big-Oh) notation for describing an algorithm’s resource (space or time) usage</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Exams</td>
</tr>
<tr>
<td>analyze the performance of simple operations on fundamental data structures</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Exams</td>
</tr>
<tr>
<td>choose an appropriate fundamental data structure for a given computational task</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Project, Exams</td>
</tr>
<tr>
<td>make appropriate use of existing (API library) code in solving a computational problem</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>HW, Project</td>
</tr>
<tr>
<td>develop a sound object-oriented design for moderately sized program</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Project</td>
</tr>
</tbody>
</table>
Computing Resources
You will be provided with a CSE undergraduate computing account. You may use the undergraduate lab facilities in Baldy 21. These facilities are available for use as listed on the course website. They are on card-access - use your UB card to open the door. For your own safety, and to protect the equipment in the lab, do not open or hold the door open in order to allow other people to gain entry to the lab. All students authorized to use the lab have card access.

You are expected to become proficient at using the machines in the lab, the Linux operating system, the Java compiler as integrated with the Eclipse IDE, and whatever other software development tools the course requires you to use. Information about the CSE computing environment can be found at https://wiki.cse.buffalo.edu/services/

You are expected to use your UB e-mail account for all communications with course staff. Always include your full and an informative subject line for your e-mail.

Recitation quizzes and homeworks will be submitted via AutoLab: https://autograder.cse.buffalo.edu/auth/users/sign_in

Course Requirements
Lectures
For all but a few lectures, students will have assigned readings from the textbook and/or additional webpages. These readings are listed on the course webpage. Lectures will expand and reinforce material from those readings with an assumption students have completed it. A problem based on the material from the readings and lecture will be included on the weekly homework so students can be certain they understand this material or know when they should seek additional help. If students understand the concepts presented in the readings and course slides (and their own notes), students should be able to complete each homework problem in under 15 minutes.

Class attendance is mandatory; this policy discusses enforcement. If you miss a class, you are responsible for talking to your classmates, TAs, or the instructor to find out what happened. If you must miss class for an extended period of time, notify your instructor as soon as possible, and see your instructor immediately upon your return in order to determine how to catch up. If you have missed a significant portion of the semester, it may be recommended that you resign from the course.

Recitations
The recitations are an integral part of the course. Recitation sections are held in the computer lab in Bell 340. Recitations will begin with the (group-based) recitation activity and then continue with an (individual) recitation quiz. To receive credit, you must be in Bell 340 during the recitation in which you are registered. Your attendance at every recitation is critical.

The recitations often ask you to apply the material that was presented in class over the past week. Applying a new skill is a key skill in computer science, but also one of the most difficult. Recitations therefore provide an excellent forum to get individualized attention to your questions. Some material needed for the course project will only be covered in recitations.

Time outside of class
Office hours are another chance for you to get individualized answers to your questions about the course. Both the instructor and the teaching assistants have scheduled office hours posted on the course website. Office hours are held on a first-come first-served drop-in basis. No appointment is necessary, but be aware that office hours become increasingly busy as deadlines or exams approach. Plan your use of office hours accordingly. Meetings outside of planned office hours are possible, but if you want to meet outside of planned office hours, you will need to talk to the instructor or teaching assistant and see if their schedule allows.

New York State Board of Regents regulations specify for every hour of time spent in lecture or recitation, students are expected to spend 2–3 hours on work outside of class. This means that you should expect to spend 8 – 12 hours each week on readings, homework, projects, and studying.
Grading Policy

Each student’s grade is computed from a weighted average of the following items:

<table>
<thead>
<tr>
<th>Item</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Final Exam</td>
<td>25%</td>
</tr>
<tr>
<td>Tests</td>
<td>25%</td>
</tr>
<tr>
<td>Homework Problems</td>
<td>10%</td>
</tr>
<tr>
<td>Adult Learning Questions</td>
<td>8%</td>
</tr>
<tr>
<td>Recitation Activities</td>
<td>5%</td>
</tr>
<tr>
<td>Recitation Quizzes</td>
<td>17%</td>
</tr>
<tr>
<td>Course Project</td>
<td>10%</td>
</tr>
</tbody>
</table>

If necessary, the instructor may revise how the final grade will be calculated. In this situation, changes will be announced during lecture and an announcement will be sent via e-mail to each student’s UB e-mail account. Each item within the course grade is described below.

The final letter grade is based upon the following cutoffs:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Cutoff</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>93+</td>
</tr>
<tr>
<td>A-</td>
<td>90-92</td>
</tr>
<tr>
<td>B+</td>
<td>87-89</td>
</tr>
<tr>
<td>B</td>
<td>83-86</td>
</tr>
<tr>
<td>B-</td>
<td>80-82</td>
</tr>
<tr>
<td>C+</td>
<td>77-79</td>
</tr>
<tr>
<td>C</td>
<td>73-76</td>
</tr>
<tr>
<td>C-</td>
<td>70-72</td>
</tr>
<tr>
<td>D+</td>
<td>67-69</td>
</tr>
<tr>
<td>D</td>
<td>60-66</td>
</tr>
<tr>
<td>F</td>
<td>0-59</td>
</tr>
</tbody>
</table>

If necessary, the instructor may revise these cutoffs downward.

Tests (25% of total course grade) & Final Exam (25% of total course grade)

This class will include two tests. These tests are held outside of regularly scheduled lectures. Students will need to show a valid photo ID (preferably their UB Card) to be allowed into each test. The tests are scheduled for:

- Test #1 - Tuesday 10/9 from 8:30PM – 9:30PM in Knox 109/110
- Test #2 - Tuesday 11/6 from 8:30PM – 9:30PM in Knox 109/110

Room assignments and the exact times of these tests will be announced on the course website and in an e-mail sent to each student’s UB account prior to each test. Each test examination is worth 12.5% of your final course grade.

A comprehensive final examination will be given during the final exam period. The final exam is worth 25% of your final course grade. The university sets the day, time and location for this examination. It is YOUR RESPONSIBILITY to check the HUB to find the exam info. Exam times can move, so you plan to be available for the entire examination period. To find your examination schedule, see:

http://registrar.buffalo.edu/schedules/finalexams.php

In accordance with the University Catalog, if you miss an examination because of sickness or similar reasons, you will be required to notify the instructor in writing and provide a written note from a detailing the period during which you were medically incapable of taking the exam is required. Notify me as early as possible in writing (e-mail is acceptable); missing an examination for an expected absence will not be excused unless you received the instructor’s permission prior to the exam.

An unexcused absence from an examination will result in a grade of 0 for that examination. Make-up examinations are only available for excused absences.

If the average of your test average and final exam score is below a passing grade (“D” or higher), you will automatically earn an “F” in the course.
Homework Problems (10% of total course grade)
Students will be assigned a set of homework problems to be completed each week. The homework problems in the set will correlate with lectures – the first problem will be on material from the lecture on Friday, the second problem from the lecture on Monday, and so on. For the first week of classes, the problems will be due on Thursday, Sept. 6th at 11:59PM; all other problem sets will be due on Fridays at 12:45PM. Late submissions will NOT be accepted. While handed out and due as a set, each problem is graded individually. There will be 34 problems assigned over the semester. This component’s score will be calculated by averaging the top 31 homework grades a student has earned.

Adult Learning Question (8% of final course grade)
Questions will be asked regularly in lecture and students will need to respond using the TopHat system. Most lectures will start with 2 or 3 questions reviewing vocabulary or basics from the previous lecture. These questions will be asked only once and correct answers on these questions will be worth 1 point. On questions asked during the lecture, students will be polled twice for their answer. The first polling is ungraded, but is only getting students to think about their answer. The second time, students answers will receive up to 2 points. These points can be awarded for both participation and correctness. Questions labeled “reinforce” (which have students use ideas that we just discussed) will see 2 points earned for correct answers and 0 points for an incorrect answer. “Application” questions (which ask students to apply course topics in a new or different approach) will award 2 points for students submitting a correct answer and 1 point for an incorrect submission. “Tricky” questions will push students in new directions and simply award 2 points for any student submitting an answer.

Missed questions cannot be made up, but reflecting the educational nature of these questions, the component will be graded using 85% of the maximum score. As an example, if there were 50 questions students could earn up to 100 points, but their score will be calculated by dividing their total points earned by 85 (e.g., 100 * 0.85).

Allowing another student to use your TopHat account is an academic integrity policy violation by both the student whose account is being used and the student answering the questions.

Recitation Activities (5% of total course grade)
Most week’s recitations will start with each project group completing a paper-and-pencil problem set. These problem sets will help reinforce the topics discussed in the previous week’s lectures and bridge the lectures to the recitation quizzes and tests. The activities will be handed out at the start of the recitation and must be submitted back to the UTA by the time specified. To receive credit on an activity, students must be an active participant in their group’s work. Students arriving more than 5 minutes after the UTA handed out the activity, will have their grade reduced. Each of the activities will be weighted equally when calculating this portion of the final course grade.

Recitation Quizzes (17% of total course grade)
Nearly every week’s recitation includes a programming quiz. The lowest quiz grade will be dropped and the remaining quizzes are weighted equally when calculating the recitation quiz portion of your final grade. These programming quizzes are completed under controlled conditions. The quiz begins after the recitation activity and must be submitted before the time specified. Late work is not accepted and arriving late does NOT change this deadline. You must bring a valid photo ID to each recitation.

In accordance with the University Catalog, if you miss a quiz because of sickness, you will be required to notify the instructor and provide a written note from detailing the period during which you were medically incapable of completing the assignment. In these cases, see your instructor as soon as you return to class.

Missing a quiz because of an unexcused absence will result in earning a 0 on that quiz. Make-up quizzes are only available for students with documentation of an excusable absence.

Course Project (10% of total course grade)
This course has a team programming project. This project gives you experience designing and implementing moderately large piece of software in a team setting. This helps develop your design and development skills, lets you use real-world software tools, and practice working in a team. All of these skills are important for your future career. Late project submissions are not accepted.
While the course project is team-based, it is important for students to have their efforts acknowledged. After each stage of this project, team members will submit peer- and self-evaluations. The details of these evaluations will be provided later in the term, but the results of these evaluations will be factored into each student’s grade for that phase.

Incomplete (I) grades
We will follow the UB Undergraduate Catalog Statement on Incomplete Grades, found in the Undergraduate Catalog. Generally, incomplete (“I”) grades are not given. However, very rarely, circumstances truly beyond a student’s control prevents him or her from completing work in the course. In such cases the instructor can give a grade of “I”. The student will be given instructions and a deadline for completing the work, usually no more than 30 days past the end of the semester. University and department policy dictate that “I” grades can be given only if the following conditions are met:

- An Incomplete will only be given for missing a small part of the course.
- An Incomplete will only be given when the student misses work due to circumstances beyond his/her control.
- An Incomplete will only be given when the student is passing the course except for the missed material.
- An Incomplete is to be made up with the original course instructor within the time specified by the appropriate University regulation (see appropriate document above), and usually within the following semester.
- An Incomplete will not be given to allow the student to informally retake the entire course, and have that grade count as the grade of the original course.

Incomplete grades cannot be given as a shelter from poor grades. **It is your responsibility to make a timely resignation from the course if you are doing poorly for any reason.** The last day to resign the course is Friday, November 11, 2016.

Accessibility Resources
25 Capen Hall, Tel: 645-2608, TTY: 645-2616, Fax: 645-3116  www.buffalo.edu/accessibility
If the Accessibility Resources office has determined that you are eligible for class accommodations, such as recruiting note-takers, readers, or extended time on exams or assignments, you must provide the course instructor with written documentation before any accommodation can be provided.

Counseling Center
120 Richmond Quad, Tel: 645-2720, Fax: 645-2175  ub-counseling.buffalo.edu
The Counseling Center staff is trained to help you deal with a wide range of issues, including how to study effectively and how to deal with exam-related stress. Services are free and confidential.

Distractions in the Classroom - Behavioral Expectations – UB Policy
Classroom "etiquette" expectations include:

- Attending classes and paying attention. Do not ask the instructor in class to go over material you missed by skipping a class or not concentrating;
- If you must enter a class late, do so quietly and do not disrupt the class by walking between students and the instructor. Do not leave class unless it is an absolute necessity and then leave by the least disruptive means possible;
- Not talking with other classmates while the instructor or another student is speaking;
- If you have a question or a comment, please raise your hand, rather than starting a conversation about it with your neighbor;
- Turn off your cell phones, pagers, and other noise making devices. If you need to keep the device on, place it in silent/vibrate mode so it will not disturb the class;
- Avoid audible and visible signs of restlessness. These are both rude and disruptive to the rest of the class;
- Focus on class material during class time. Sleeping, talking to others, doing work for another class, reading the newspaper, checking email, and exploring the internet are rude and unacceptable;
- Not packing bookbags or backpacks until the instructor has dismissed class.
Academic Integrity

Source: http://www.cse.buffalo.edu/undergrad/policy_academic.php

The academic degrees and the research findings produced by our Department are worth no more than the integrity of the process by which they are gained. If we do not maintain reliably high standards of ethics and integrity in our work and our relationships, we have nothing of value to offer one another or to offer the larger community outside this Department, whether potential employers or fellow scholars.

For this reason, the principles of Academic Integrity have priority over every other consideration in every aspect of our departmental life, and we will defend these principles vigorously. It is essential that every student be fully aware of these principles, what the procedures are by which possible violations are investigated and adjudicated, and what the punishments for these violations are. Wherever they are suspected, potential violations will be investigated and determinations of fact sought. In short, breaches of Academic Integrity will not be tolerated.

Departmental Statement on Academic Integrity in Coding Assignments and Projects

All academic work must be your own. Plagiarism, defined as copying or receiving materials from a source or sources and submitting this material as one's own without acknowledging the particular debts to the source (quotations, paraphrases, basic ideas), or otherwise representing the work of another as one's own, is never allowed. Collaboration, usually evidenced by unjustifiable similarity, is never permitted in individual assignments. Any submitted academic work may be subject to screening by software programs designed to detect evidence of plagiarism or collaboration.

It is your responsibility to maintain the security of your computer accounts and your written work. Do not share passwords with anyone, nor write your password down where it may be seen by others. Do not change permissions to allow others to read your course directories and files. Do not walk away from a workstation without logging out. These are your responsibilities. In groups that collaborate inappropriately, it may be impossible to determine who has offered work to others in the group, who has received work, and who may have inadvertently made their work available to the others by failure to maintain adequate personal security. In such cases, all will be held equally liable.

Departmental Policy on Violations of Academic Integrity

The CSE Department has a zero-tolerance policy for AI violations.

All AI violations will be reported to the department, school, and university, and recorded.

Even a 1st offense will receive an "F" for the course unless the instructor finds there are mitigating factors that make it appropriate to reduce the penalty. Under departmental policy, subsequent AI violations must result in an "F" grade, with no exceptions for the form or course in which the earlier violation occurred.

Course Policy on Violations of Academic Integrity

Aside from recitation activities and the course project, work you submit for credit in this class is individual work.

Between instructor office hours, UTA office hours, Piazza, e-mail, and reviewing the textbook and slides, students have sufficient resources to have their questions answered. There is no reason why students should violate AI policies. While it a reduced sanction from the and so it is extremely unlikely that the instructor will deem it appropriate to reduce the sanction resulting from an AI policy violation.