

**University at Buffalo**  
*Department of Computer Science & Engineering*  
338 Davis Hall – (716) 645-3180

## Syllabus

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

### Instructor

Name	Office	Phone	Email	Office hours
Dr. Carl Alphonc	343 Davis	645-4739	alphonc@buffalo.edu	T 9:00 – 11:50 R 12:00 – 1:50 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:

<http://www.cse.buffalo.edu/faculty/alphonc/Courses/Fall2014/cse116>

Lecture Times:

A MWF 1:00 PM – 1:50 PM Knox 110

Recitation Times (all recitations meet in Baldy 21):

A1 W 3:00 PM - 4:50 PM

A2 T 11:00 AM - 12:50 PM

A3 R 11:00 AM - 12:50 PM

A4 R 5:00 PM - 6:50 PM

A5 M 6:00 PM - 7:50 PM

A6 F 6:00 PM - 7:50 PM

### 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.

## Objectives and 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 has a significant relationship with the following program objectives:

(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.*

The course has a strong relationship with the following program objectives:

(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 has a significant relationship with the following program objectives:

(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.*

The course has a strong relationship with the following program objectives:

(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.*

## Learning Outcomes and Program Objectives 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.

Upon successful completion of this course a student will be able to...	CEN-a	CEN-h	CEN-k	CS-a	CS-c	CS-d	CS-i	Assessment Instruments
demonstrate the operation of fundamental data structures	✓		✓	✓	✓	✓	✓	HW, Project, Exams
implement fundamental data structures in Java	✓		✓	✓	✓	✓	✓	Project
write unit tests from informal specifications		✓	✓		✓	✓	✓	Project, Exams
execute unit tests using the JUnit unit testing framework		✓	✓		✓	✓	✓	HW, Project
employ a versioning tool (CVS) to manage program files for teamwork		✓	✓		✓	✓	✓	Project
use an integrated development environment (such as Eclipse's JDT) to perform typical program editing and navigation tasks		✓	✓		✓	✓	✓	HW, Project

define the O (big-Oh) notation for describing an algorithm's resource (space or time) usage	✓	✓	✓	✓			✓	Exams
analyze the performance of simple operations on fundamental data structures	✓	✓	✓	✓			✓	Exams
choose an appropriate fundamental data structure for a given computational task	✓		✓	✓	✓	✓	✓	Project, Exams
make appropriate use of existing (API library) code in solving a computational problem			✓		✓	✓	✓	HW, Project
develop a sound object-oriented design for moderately sized program			✓		✓	✓	✓	Project

## 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

A Course Response System, the ResponseCard NXT (commonly called a “clicker”), is required. Register your clicker in UBLearn. See <http://www.buffalo.edu/ubit/service-guides/teaching-technology/learning-resources-for-students/ubclicks.html> for more information.

There is one required textbook for this course.

- William J. Collins. 2011. *Data Structures and the Java Collections Framework* (3<sup>rd</sup> ed), John Wiley & Sons. (ISBN: 978-0-470-48267-4).

You may find the following books useful as well. They are *not* required.

- Kent Beck with Cynthia Andres. 2005. *Extreme Programming Explained* (2<sup>nd</sup> ed), Pearson Education (Addison-Wesley). (ISBN-13: 978-0-321-27865-4).
- Cay Horstmann. 2006. *Object-Oriented Design and Patterns* (2<sup>nd</sup> ed), John Wiley & Sons, Inc. (ISBN-13: 978-0-471-74487-0)
- Cay S. Horstmann and Gary Cornell. 2005. *Core Java 2, Volume I: Fundamentals* (7<sup>th</sup> ed), Prentice Hall. (ISBN: 0-13-148202-5)

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

## 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 who are authorized to use the lab will be granted card access.

The name of the server that you will be connecting to in the lab is `styx.cse.buffalo.edu`. You have the ability to connect to other CSE department servers remotely from other sites, on or off campus.

You are expected to become proficient at using the machines in the lab, the Linux operating system, the Java compiler as integrated with Eclipse IDE, and whatever other software development tools the course requires you to use. It is your responsibility to ensure that any programs you write for this course compile

using the Java compilers installed on the department's machines. Information about the CSE computing environment can be found at,

<https://wiki.cse.buffalo.edu/services/>

Any e-mail communication that you send regarding this course must be sent from your UB e-mail account. Under no circumstances will e-mail from non-UB accounts be acknowledged or answered. You must include an informative subject line in all e-mail, and include your full name in any e-mail correspondence.

You are also required to read mail sent to your UB e-mail account. All e-mail that we send to you will be sent to your UB e-mail account only. Our feedback on materials you hand in electronically will be available either through Web-CAT,

<https://web-cat.cse.buffalo.edu:8443/Web-CAT>

or by e-mail sent to your UB e-mail account. Since you may request re-grades of work only within a set period from the time that the feedback was provided to you, it is in your best interest to read your UB e-mail account on a daily basis.

## Course Organization

The course has both a lecture component and a recitation (lab) component. Each component plays a role in helping you achieve the objectives of the course. **If you do not participate fully in both you should not expect to do well in the course.** Outside of the scheduled course times, both office hours and your own study times are critical components of the course.

### *Lectures*

The conceptual and theoretical course content will be delivered primarily in the lectures, complemented by readings from the textbook(s). You must review readings prior to attending a lecture, and you are expected to review the readings again, along with any notes you took, after the lecture.

Some of the topics will be difficult. It is therefore absolutely essential that you ask questions whenever something is said which you do not understand.

*Attendance in all lectures is critical to your success in this course. Lack of attendance may lower your course grade.* If you are unable to attend a lecture because of sickness or similar reasons, make sure you get the notes from a classmate. If you are out of class for an extended period of time because of sickness, 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 is recommended that you resign from the course.

### *Recitations*

The recitations are an integral part of this course. *Attendance in all recitations is critical to your success in the course. Lack of attendance may lower your course grade.*

Recitations do not meet in the first week of classes. Recitations meet for the first time the second week of classes.

*Time outside of class***Office hours**

Office hours offer you the opportunity to ask more individual questions about the course material than can typically be addressed in lecture. Both the instructor and the teaching assistants have scheduled office hours. Office hours are held on a first-come first-served drop-in basis. No appointment is necessary to attend office hours. Be aware that office hours become increasingly busy the closer it is to a deadline or exam. Plan your use of office hours accordingly.

Individual appointments may be arranged, if needed, as schedules allow.

**Study time**

In this course, as in any course, you are expected to put in additional time beyond the scheduled class times. Professors generally expect that for each credit hour a class carries a typical student will put in 2 – 3 hours of time **each week** outside of class. **Since this is a 4 credit course that translates into 8 – 12 hours of time outside of lecture and recitation times, each week.** During this time you should review your lecture notes, attend office hours as needed, get hands-on practice applying the concepts and theoretical constructs discussed in class, prepare to code up your homework, and (later in the semester) arrange for meetings with project teammates to work on the course project.

As a rough guide, you should expect to spend **at least** the following time working on this course, **each week**:

lectures (3 hours) – recitation (2 hours) – programming assignments/project (4 hours) – individual study (4 hours)

**Grading**

The following indicates the grade breakdown which will be used in assigning grades in the course. I reserve the right to make adjustments if I deem it to be necessary. Any changes will be communicated to the class in writing via e-mail to each student's UB e-mail account.

***Exam component (50% of final course grade)***

There will be three examinations given during the semester and one final examination at the end of the term. You must attend all examinations.

The **two in-semester (unit) exams** will be held as follows, unless otherwise announced on the course website:

- EXAM #1 – **Thursday 10/9, in the evening**
- EXAM #2 – **Thursday 11/6, in the evening**

The **comprehensive final examination** will be given during the final exam period. The university schedules final examinations. It is YOUR RESPONSIBILITY to check the HUB for the date, time and place of the final exam. See <http://blogs.advising.buffalo.edu/beadvised/have-you-checked-your-final-exam-schedule>

**Since the exam schedule can change, do not make plans to travel during the examination period.**

If you miss an examination because of sickness or similar reasons, written notice and acceptable written documentation must be provided, as specified in the University Catalog (Class Attendance, page 25). In the case of illness, you **MUST** visit a physician and obtain a note detailing the period during which you were medically incapable of taking the exam. Notify me as early as possible in writing (e-mail is acceptable) if you miss an exam, before the exam takes place unless medically impossible. See your instructor as soon as you return to class.

**If you miss an examination without a valid excuse, you will receive a zero grade for that examination. No make-up examinations will be available without a valid excuse.**

You must bring a valid form of picture ID with you to each examination (a UB Card is preferred).

There are two options for calculating your score for the exam component of the course:

	Option #1	Option #2
In-semester exams	25%	0%
Final Exam (Cumulative)	25%	50%

The option which gives you the highest score in the course will be used automatically.

**You must attempt all in-semester exams in order for the final-exam only option to be available to you. If you do not write all in-semester exams, you cannot make use of the final-exam only option.**

The motivation for having two grading options available is to ensure that you are not penalized if you had a rough start in the course, but managed to do really well on the final exam. If you do poorly on the in-semester exams, you can still do well in the course by demonstrating that you have learned the material on the final exam. Of course, if you do poorly on the in-semester exams, this means you are playing without a safety net.

A necessary *but not sufficient* condition for receiving a passing grade in the course is having a passing exam component grade. See the table mapping numeric grades to letter grades later in the syllabus for the passing grade. **In other words, if your exam component score is not passing, you will fail the course, no matter how good your overall course grade is.**

### ***Individual programming component (20% of final course grade)***

Individual programming assignments will be given throughout the semester, on a roughly weekly basis. All individual programming exercises will count equally. Each individual programming assignment write-up will be done in recitation, individually, under controlled conditions. The write-ups start promptly at the start of recitation, and finish at a set time. The end time will not be adjusted to accommodate late-comers.

A necessary *but not sufficient* condition for receiving a passing grade in the course is having a passing individual programming component grade. See the table mapping numeric grades to letter grades later in the syllabus for the passing grade. **In other words, if your individual programming component score is not passing, you will fail the course, no matter how good your overall course grade is.**

If you miss an individual programming assignment write-up because of sickness or similar reasons, *visit a physician and obtain a note detailing the period during which you were medically incapable of taking doing the write-up*. Notify your instructor immediately via e-mail or telephone (voice mail) if you are going to miss an individual programming assignment write-up, before it takes place unless medically impossible. See your instructor as soon as you return to class.

**If you miss an individual programming assignment write-up without a valid excuse, you will receive a zero grade for that assignment. No make-up will be available without a valid excuse.**

You must bring a valid form of picture ID with you to each programming assignment (a UB Card will

suffice).

***Team programming component (12% of final course grade)***

This course has a team programming project. Designing and implementing this project will give you experience working with a moderately large piece of software, which is very important in order to develop sound design and programming skills and habits. You will work as a member of a small team to complete the course project. The course project will be due no later than 9:00 PM on 5/9, unless adjusted on the course website. The project will be submitted on-line.

**Late project submissions will not be accepted under any circumstances.**

**Peer evaluation**

The programming project is a team-based project. For each stage submission each team member will evaluate the contribution of each member of their team, including themselves. Details on the peer evaluation procedure will be handed out with the project description. I reserve the right to ignore the peer evaluations in cases where I feel they are not an accurate reflection of a student's contribution to the project.

***Professionalism component (4% of final course grade)***

You are expected to demonstrate professionalism in all you do. This includes arriving for class on time, being attentive in lecture, attending appointments that you make, not using your cell phone (or similar devices) during lecture, addressing the course staff with respect, etc. If you consistently exhibit unprofessional behavior your professionalism grade will be reduced.

***Active Learning component (14% of final course grade)***

Questions will be asked regularly in lecture and/or recitation, for which answers must be provided using each student's individual clicker device. A correct answer earns 2 points. An incorrect answer earns 1 point. No response results in zero points.

Missed questions cannot be made up, under any circumstances. However, each student's lowest 8 question responses will be dropped from grade calculations.

Using another student's clicker is a violation of academic integrity. Allowing another student to use your clicker is a violation of academic integrity.

***Overall course grade***

Your overall course grade is determined as follows:

**If you fail *either* the exam or the individual programming component of the course, you will earn a failing grade in the course as a whole.**

**If you pass *both* the exam and the individual programming components then your course grade is the weighted sum of all the component grades.**

The following table indicates the number to letter grade mapping I will use to assign letter grades at the end of the course. I reserve the right to make adjustments to the letter grade cut-offs. If any adjustments are made cut-offs will only be adjusted lower (e.g. the cut-off for an A may be moved from 90 to 89), never higher. The Grade points column is included for your convenience only, and is not official information. The official mapping can be found in the Undergraduate Catalog. Note that the minimum passing grade, for the course as a whole and also for any course component, is 55% (unless adjusted).

Percentage score	Letter grade	Grade points
90-100	A	4.0
85-89	A-	3.67
80-84	B+	3.33
75-79	B	3.0
70-74	B-	2.67
65-69	C+	2.33
60-64	C	2.0
55-59	C-	1.67
0-54	F	0.0

## Regrading

If you have a question about the grading of any piece of work, first consult with the teaching assistant who graded your work. If you cannot resolve your questions with the teaching assistant, you should consult with the course instructor.

Any questions about the grading of a piece of work ***must be raised within one week of the date that the work was returned*** by Web-CAT, the teaching assistant or the instructor. Active learning responses cannot be regarded. It is your responsibility to ensure that you bring your clicker to class, that you set it to the correct channel, and that the batteries have sufficient power.

In other words, if you do not pick up your work in a timely fashion, you may forfeit your right to question the grading of your work.

## 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.



Incompletes 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 7, 2014.**

### **Accommodation via Accessibility Resources**

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.

Accessibility Resources is located at 25 Capen Hall, Tel: 645-2608, TTY: 645-2616, Fax: 645-3116,  
**<http://www.buffalo.edu/accessibility>**

### **Counseling Center**

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.

The Counseling Center is located at 120 Richmond Quad , Tel: 645-2720, Fax: 645-2175,  
**<http://ub-counseling.buffalo.edu>**

### **General Notes**

If you don't understand something covered in class, ask about it right away. The only silly question is the one not asked. If you get a poor mark on an assignment, quiz, or exam, find out why right away. Don't wait a month before asking. The instructor and teaching assistants are available to answer your questions. Don't be afraid to approach the instructor or T.A. in class, during office hours, or through e-mail.

## **Distractions in the Classroom - Behavioral Expectations**

***Unless explicitly authorized, all electronics (aside from “clickers”) must be turned OFF and be PUT AWAY. This includes cell phones, tablets and laptops. I have adopted this policy based on student feedback.***

The following is the text of a policy adopted by the Faculty Senate on 5/2/2000. You are expected to know and adhere to this policy.

### **OBSTRUCTION OR DISRUPTION IN THE CLASSROOM – POLICIES UNIVERSITY AT BUFFALO**

To prevent and respond to distracting behavior faculty should clarify standards for the conduct of class, either in the syllabus, or by referencing the expectations cited in the Student Conduct Regulations.

Classroom "etiquette" expectations should include:

- Attending classes and paying attention. Do not ask an instructor in class to go over material you missed by skipping a class or not concentrating.
- Not coming to class late or leaving early. If you must enter a class late, do so quietly and do not disrupt the class by walking between the class and the instructor. Do not leave class unless it is an absolute necessity.
- 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.
- Showing respect and concern for others by not monopolizing class discussion. Allow others time to give their input and ask questions. Do not stray from the topic of class discussion.
- Not eating and drinking during class time.
- Turning off the electronics: cell phones, pagers, and beeper watches.
- Avoiding audible and visible signs of restlessness. These are both rude and disruptive to the rest of the class.
- Focusing 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 unacceptable and can be disruptive.
- Not packing bookbags or backpacks to leave until the instructor has dismissed class.

## Academic Integrity

*Source: [http://www.cse.buffalo.edu/undergrad/policy\\_academic.php](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***

The following statement further describes the specific application of these general principles to a common context in the CSE Department environment, the production of source code for project and homework assignments. It should be thoroughly understood before undertaking any cooperative activities or using any other sources in such contexts.

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.

These policies and interpretations may be augmented by individual instructors for their courses. Always check the handouts and web pages of your course and section for additional guidelines.

### ***Departmental and Course Policy on Violations of Academic Integrity***

If, after following the procedures required by the University for investigation of suspected breaches of academic integrity, a student is found guilty, the policy of the department of Computer Science & Engineering is that the student minimally receive a grade of F in the course. My course policy is that the standard consequence of a violation of academic integrity is failure in the course.