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 Alphonce	343 Davis	645-4739	alphonce@buffalo.edu	Tuesday 10:00 AM – 12:00 PM Wednesday 4:00 PM – 5:00 PM Friday 11:00 AM – 12:00 PM OR e-mail for appointment

Teaching Assistants

See course website.

Course Information

Credit hours:

CSE 115 Introduction to Computer Science for Majors I – 4 credits
CSE 503 Computer Science for Non-Majors I – 3 credits

`http://www.cse.buffalo.edu/faculty/alphonce/cse115/` **`ixlearn.com/dashboard/SLYMNE`**

Course Description

This course provides the fundamentals of the field to computer science and computer engineering majors, introducing students to algorithm design and implementation in a modern, high-level programming language, with an emphasis on problem-solving by abstraction. Topics include object-oriented design using a formal modeling language; fundamental object-oriented principles such as classes, objects, interfaces, inheritance and polymorphism; simple event-driven programming; data types; variables; expressions; basic imperative programming techniques, including assignment, input/output, subprograms, parameters, sequencing, selection and iteration; the use of aggregate data structures, such as arrays or more general collections; simple design patterns.

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

Student Learning Outcomes

This course adheres closely to the recommendations of ACM's CC2001 curriculum document for a first semester introductory course. It covers topics from the following knowledge units: **PF1** Fundamental programming constructs, **PF2** Algorithms and problem-solving, **PF5** Event-driven programming, **PL1** Overview of programming languages, **PL4** Declarations and types, **PL5** Abstraction mechanisms, **PL6** Object-oriented programming, **HC2** Building a simple graphical user interface, **SE1** Software design, **SE2** Using APIs, **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-e) An ability to identify, formulate, and solve hardware and software computer engineering problems using sound computer engineering principles
(CEN-g) An ability to effectively communicate technical information in speech, presentation, and writing.
(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-f) An ability to communicate effectively with a range of audiences.
(CS-i) An ability to use current techniques, skills, and tools necessary for computing practice.

Student Learning Outcomes mapping

Upon successful completion of this course a student will be able to…	*(CEN-a)*	*(CEN-e)*	*(CEN-g)*	*(CEN-k)*	*(CS-a)*	*(CS-f)*	*(CS-i)*	*Assessment Instruments*
write software that uses object-oriented design and its key concepts of encapsulation, inheritance, and polymorphism	✔	✔	✔	✔	✔		✔	Labs, Exams
create an event-driven program that has a graphical user interface (GUI) and components that the user of the program can interact with (menus, buttons, etc)	✔	✔	✔	✔	✔		✔	Labs, Exams
interpret and create UML class diagrams that reflect the design of a piece of software	✔	✔	✔	✔	✔	✔	✔	Labs, Exams
interpret and create UML object diagrams that reflect the runtime behavior of software	✔	✔	✔	✔	✔	✔	✔	Exams
explain the difference between stack allocation and heap allocation of memory	✔	✔	✔	✔	✔	✔	✔	Exams
define standard computing vocabulary, such as scope, lifetime, class, object, method, parameter, etc.	✔	✔	✔	✔	✔	✔	✔	Exams
identify where and when to use constructs such as selection and iteration in their programs	✔			✔	✔		✔	Labs, Exams
use an integrated development environment (such as Eclipse's JDT) to perform typical program editing and navigation tasks	✔			✔	✔		✔	Labs
use various types of collections and their iterators in solving problems involving multiple objects of the same type	✔			✔	✔		✔	Labs, Exams

Prerequisites

There are no formal UB course prerequisites for this course. You do not need previous programming experience to take this course, but you must have completed high school pre-calculus (algebra and trigonometry) coursework, and be ready to take calculus. **If you are currently taking ULC147, ULC148 or MTH115 you have NOT satisfied this prerequisite!**

Textbook

We are using an on-line textbook whose cost is $67. Sign up at https://zybooks.zyante.com/ using code **BUFFALOCSE115AlphonceFall2016**

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.

Our feedback on materials you hand in electronically will be available either through Web-CAT,

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

Course Requirements

The course has both a lecture component and a recitation (lab) component. **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 lecture.

Attendance in all lectures is critical to your success in this course. If you are unable to attend a lecture because of sickness or similar reasons, get 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.

Labs

The labs are an integral part of the course. *Attendance in all labs is critical to your success in the course.* In each lab section, the TAs will cover material pertinent to the current assignment. The lab sections are held in the computer lab in Baldy 21 and you will have the opportunity to work on your assignments during recitation and ask questions of the TA about your work.

The labs may also review and extend lecture material and are also an excellent forum for asking more individual questions about the course material than can typically be addressed in lecture. Some material needed to do the programming labs will be covered only in lab.

Time outside of class

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. 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.

In this course, as in any course, you are expected to put in additional study 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*.** 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 (4 hours) – individual study (4 hours).

Grading Policy

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 (40% of final course grade)

There will be **two unit examinations** given during the semester and **one final examination** at the end of the term. You must bring a valid form of picture ID with you to each examination (a UB Card is preferred).

The **first unit exam** will be held on Tuesday October 4 at 8:45 PM and the **second unit exam** will be held on Tuesday November 15 at 8:45 PM. Room assignments will be announced in the week prior to each exam.

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. 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.

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.

There are two options for calculating your score for the exam component of the course (the option which gives you the highest score in the course will be used automatically):

	Option #1	Option #2
Unit exams	20% (10% each)	0%
Final Exam (Cumulative)	20%	40%

You must attempt all unit exams in order for the final-exam only option to be available to you. If you do not write all unit 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.

Lab component (40% of final course grade)

There will be weekly programming labs, **weighted equally**. There will likely be eleven labs. The purpose of these is to reinforce and deepen your understanding of the broader concepts discussed in class through application of those concepts to concrete problems. The programming labs are designed to give you hands-on experience analyzing problems, developing solutions to them, and implementing these solutions in Java. These labs also serve to give you feedback on your understanding of the material. Labs will be due no later than 9:00 PM on the night before your next recitation session. Labs are submitted on-line.

Early policy for lab submissions

Any programming lab whose final submission is early will have a 1 point bonus (out of 100) awarded per day (24 hours) early. The maximum bonus on any one submission is 5 points.

Late policy for lab submissions

Any programming lab submission which occurs after the due date is considered late, and will have a 20 point penalty (out of 100) imposed per day (24 hours), or portion thereof, late. In no case shall a lab grade be less than zero. No exceptions will be made to this policy, except in those cases where written notice and acceptable written documentation is provided, as specified in the University Catalog (Class Attendance).

Active Learning component (20% of final course grade)

There are two elements to the active learning component. One is questions asked in class (lecture or recitation), the other is exercises from the on-line textbook. Details of grading this component will be given in lecture.

Overall course grade

Your overall course grade is determined as follows:

If you *fail either* the exam *or* the lab component of the course, you will fail the course as a whole.

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

Note that your lab grade cannot count for more than 40%, your exam grade cannot count for more than 40%, and your active learning grade cannot count for more than 20%. This means, for example, that your lab bonuses can only help you reach the maximal lab grade, but can never help your exam grade.

The table below gives the number to letter grade mapping for the course. I reserve the right to adjust the cut-offs. 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 minimum passing grade for the course as a whole and also for any course component is 55% (unless adjusted).

Percentage score	Letter grade	Percentage score	Letter grade
90-100	A	65-69	C+
85-89	A-	60-64	C
80-84	B+	55-59	C-
75-79	B	0-54	F
70-74	B-		

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, ask 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. 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 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

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

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 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.

Course Policy on Violations of Academic Integrity

ALL WORK YOU SUBMIT FOR CREDIT IN THIS CLASS IS INDIVIDUAL WORK. My course policy is that the standard consequence of ANY VIOLATION of academic integrity is a grade of F in the course.