

Welcome

Please make sure you get a copy of the syllabus (extra copies are available at front).

CSE 115 – Introduction to Computer Science for Majors I

CSE 503 – Computer Science for Non-Majors I

- Instructor:
 - Dr. Adrienne Decker
 - Please call me Adrienne
 - But if you must be formal, it's Dr. Decker or Professor Decker.
 - NOT Miss Decker, or Ms Decker or Mrs. Decker
 - Office: 130 Bell Hall

CSE 115 – Introduction to Computer Science for Majors I

CSE 503 – Computer Science for Non-Majors 1

- Email: adrienne@buffalo.edu
 - Email you send me should be from your UBIT email account and should include your <u>full name</u> and <u>which course you</u> <u>are taking</u>.

CSE 115 – Introduction to Computer Science for Majors I and CSE 503 – Computer Science

for Non-Majors .

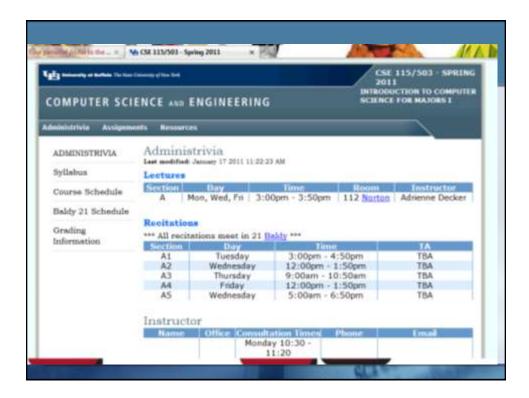
- Office Hours:
 - Monday 10:30–11:20
 - Wednesday 10:00–11:20 and 2:00–2:50
 - Friday 2:00–2:50

Information on the Web

All course content will be on the course website:

http://www.cse.buffalo.edu/faculty/adrienne/SP2011/cse11 5

UBLearns will be used to post course grades



Course Structure

- Lecture (3 hours each week)
- Recitation (2 hours each week)
 - You should be registered for a recitation section and a lecture.
 - Recitations do not meet this week.
- If you are interested in changing your recitation section and are unable to do so through the registration system, please fill out Recitation Change Request Form.

Course Description and Prerequisites

- This course is an introduction to computer science for intended computer science and computer engineering majors.
- There are no prerequisites for this course, but you should have some familiarity with a computer (that is, you should have used one before). There is a co-requisite of MTH 141.

Textbook

- Textbook will be available through course website electronically.
- You can print it out or view it online.

Computing Resources

- Projects for this course will be completed on the department of Computer Science & Engineering's computer systems.
- You will be receiving an account on these systems.

Course Grades

- 56% Exams
 - Five in-class exams and one final exam. If you take all five in-class exams, the lowest of the five will be dropped and you have the following grading option available:

		Option #2
In-class Exams	28%	0%
Final Exam	28%	56%

Course Grades

- 44% Programming Assignments
 - There will be 8 assignments given throughout the semester. Weightings for each assignment are not equal (see chart in syllabus).
 - Submissions of programming assignments after the due date will incur a 50% penalty(of maximum score obtainable) per day late.

Course Grades

- Important Notes:
 - In order to be eligible to receive a grade higher than F in the course, you must have a passing average (> 50%) on the exams for the course as well as a passing average (> 50%) on the programming assignments for the course.

Letter Grades

- See chart in syllabus.
- There is no curve on the course grades. If your average falls in between the cutoffs, that is your grade.

Course Policies

- Re-grading any questions about graded work must be raised within one week of the return of the work.
- Incompletes we will follow the university's policy on incompletes – unless you meet the criterion, you will not get an incomplete.

Course Policies

- Disability Services If you are registered, please bring me the letter indicating your accommodations.
- Athletics If you are an athlete, please come to speak with me about how that will effect this course this semester.

Course Policies

- Disruption/Behavior in the Classroom
 - Take note of the University's policy on this issue (in syllabus)
 - Be respectful of each other

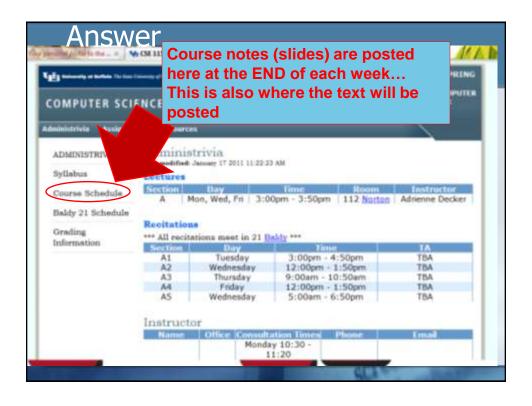
Course Policies

- Academic Integrity
 - Breaches of academic integrity will be investigated and punishments imposed in accordance with the University's policies AND my department's policies.
 - My department's policy is that ANY breach of academic integrity is punished with an F in the course (no more lenient punishments allowed).

Syllabus Confirmation

- You need to go to UBLearns and complete the syllabus confirmation "test" that is now available in the course.
- Failure to complete this "test" will render you ineligible to take the third practical exam.
- You have until the end of the day, Monday, January 31, 2011 to complete the "test".





Announcements

- Pick up (and READ) syllabus if you have not already done so.
- Syllabus Confirmation "test" on UBLearns needs to be completed by 1/31/11.
- No recitations meet this week

Course Registration

- 503 students interested in attending an alternate recitation, please fill out form.
- A6 (Thursday 4:00 5:50) has been added and is available for registration. If you are interested in attending this recitation, please change your registration through MyUB.
- If you are interested in changing to another recitation and there are open seats, please change your registration through MyUB. If there are not open seats, please fill out form.
- If you are not yet registered for this class and need to be – please see me after class.

In your notebook/on a sheet of paper

- If you are an intended CS major what is computer science?
- If you are an intended CEN major what is computer engineering?
- If you are neither what are you hoping to learn in this course/why are you taking the course?

Answers

- What is computer science?
- Study of algorithms, inner working of computer; theory behind computing & networking; how computers communicate.
 - What is computer engineering?
 - Study of how a conjuter functions 4 o penders; developing components to improve technology; Nordware/software , nk gration

So – what are they?

Well, they are all these things. There isn't a universally agreed-upon definition for these two terms, but the areas tend to overlap in many cases. In computing, your eventual job description is sometimes of your own creation.

More answers

What do you want to learn?

- Program
- Different languages of a computer
- Possible jobs/opportunities

We could say....

- Computing (or computer science) is the science of representing things inside a computer.
- One of the first types of representation people think of when they think about a computer are 0's and 1's.

Bit - binary digit Gholds either azero or a one. Bit string- A sequence of bits

So...

- We use the bits to store everything inside the computer.
- Perhaps computer science is the science of interpretation. How can we interpret the bits in meaningful ways.

How can we interpret the patterns?

Ex) 10110

-Binary number (ron-negatives)

-Twos complement (integers)

-IEEE 754 (approximate floating point numbers)

-ASCII