Syllabus:
CSE 220: Systems Programming

University at Buffalo Department of Computer Science and Engineering

Ethan Blanton

Fall 2021

Revision 2021-08-25

All students are expected to read and understand this syllabus. Failure to adhere to the policies in this syllabus may have consequences, including a negative impact on student grades, failure in the course, or administrative action against the student. *It is your responsibility to ask questions if anything in this document is unclear to you.*

This course is in-person and real-time. Students are expected to attend their assigned lecture and lab sections unless they have a University-approved reason to miss.

Instructor

Ethan Blanton
eblanton@buffalo.edu

Office Hours (subject to change; consult the course web site):

Monday	10:00–11:00	Davis 334
Thursday	13:00–15:00	Davis 334

Teaching Assistants

TA names and their office hours can be found on the course Piazza instance.

Sections

Section	Course ID	Lecture Time	Location
CSE 220 LLB-A	19959	MWF 12:40-13:30	Cooke 121
CSE 220 LLB-B	19961	MWF 15:00-15:50	Knox 110

1 Course Web Site

https://cse.buffalo.edu/~eblanton/course/cse220/

Locations, times, information regarding instructor and TA office hours, assignment deadlines, and other information can be found on the course web site.

2 Lectures and Labs

Lectures will be delivered on campus in their assigned locations. Students are expected to attend their assigned lecture unless they have a University-approved reason to miss. Attendance will not be regularly taken, but may be taken at any time at the instructor's discretion. Lecture recordings will be provided after the fact for review, or for asynchronous viewing in the event that a student must miss lecture for a University-approved absence.

Labs are at various times, as specified on HUB; every student is registered for particular a lab section, and *must attend their registered lab section* unless prior arrangements have been made with the instructor. Lecture and lab times are not scheduled by the course instructor and may be changed by the University for various reasons. Check HUB for the most recent information.

3 Course Description

CSE 220 is a 4 credit course.

> This course is an introductory course on computer systems. It introduces computer systems from a programmer's perspective, rather than a system implementer's perspective, which prepares students for more advanced topics that discuss the internals of a computer system (e.g., operating systems or computer architecture). As a result, the focus of the course is teaching programmable interfaces of a computer system as well as how to use them correctly and effectively when writing a program. The topics mainly include hardware/software interfaces (e.g., data representation in memory) and OS/application interfaces (e.g., syscalls). In discussing these topics, the course gives an overview of a complete computer system, the hardware, operating system, compiler, and network, in order to guide students through various components that modern programs rely on to accomplish their intended purposes.

4 Course Materials

There are two required texts for this course. Students are expected to have immediate access to both of these texts throughout the semester.

Title	Authors	ISBN
Computer Systems: A Programmer's Perspective (Third Edition)	Bryant and O'Hallaron	978-0-13-409266-9
The C Programming Language (Second Edition)	Kernighan and Ritchie	978-0-13-110362-7

Lecture slides will be provided electronically throughout the course of the semester.

Various readings from external sources may be assigned, in which case they will be provided or available through University resources (*e.g.*, the library or periodical subscriptions).

References to materials not required or assigned may be made, and students are encouraged to follow up on these references, but this will not be required for successful completion of the course.

5 Communication

All electronic communication from students to course staff regarding this course must occur in one of two ways:

- Messages on the course Piazza instance, or

- Email using your *official UB email account.*

For topics of a sensitive nature, please email the course instructor directly from your University-supplied email address. Emails from non-University addresses will be disregarded due to privacy concerns and FERPA regulation. For all other contacts, please do **NOT** email the course instructor directly; instead, make a private or public post to Piazza, as appropriate. Private posts of non-sensitive nature should be sent to *all course staff*. This will ensure the most timely possible response. For matters regarding a specific lab section, contact the lab assistants first, as they may be able to help resolve your issue without involving the rest of the course staff.

Students are expected to monitor the course Piazza instance and UBlearns classroom on a daily basis, checking it on every day that the University is open, as important course announcements will be posted to Piazza. Schedule changes, lab and assignment handouts, homeworks, required readings, and other materials may be posted to Piazza, and it is the student's responsibility to keep track of these things. Failure to read Piazza messages will not be accepted as an excuse for missed projects, labs, exams, or other course requirements.

Students will be added to the Piazza instance by the course instructor. If you are not, please contact the instructor by email to rectify this.

6 Prerequisites

- CSE 116. SEAS approved or Intended Majors, CS Minors.

Students are expected to have a firm understanding of the material from CSE 116 at the University at Buffalo and its prerequisites.

7 Course Requirements

The following items are required of every student, and failure to complete them may affect student grades as described in Section 9, *Grading Policy*, below.

7.1 General

Students must attend every lecture. Lecture quizzes will be given to evaluate progress in understanding lectures, but completing the quizzes without thoroughly understanding the lecture material and assigned readings will not be sufficient to ensure success on projects and exams.

Videos containing course content will be released for every lab session. Students must watch these videos and take an associated quiz. Lab section attendance is required, and students must show up for their assigned lab section unless prior arrangements have been made. Students are expected to work on the assigned lab during their assigned lab section, and the intention is that most students will complete their lab assignments during the lab section.

It will be assumed that students are familiar with all material presented in class, and any material presented in lecture or lab may appear on any test, quiz, homework assignment, or other evaluation. *Attendance and attention to lecture and lab materials are critical to success in this course.*

A quiz evaluating students' understanding of the University and Department academic integrity policies must be completed by all students, and all students must achieve 100% accuracy on this quiz.

7.2 Assignments

Several programming assignments will be required of all students. Programming assignments are intended to improve student understanding of the course material as well as demonstrate student mastery of certain core concepts.

Lab activities will be provided weekly, which may range from small programming assignments to written activities to group discussions and beyond. Students are expected to participate fully in lab and will be evaluated accordingly. Course staff will be available in labs to assist students, guide tasks, and present additional course material.

Programming assignments and lab activities, unless otherwise explicitly specified, are *individual activities*, and collaboration between students to complete any such assignment is a violation of the course academic integrity policy.

Written homework may be assigned, and students are expected to complete any such assignments in a timely fashion, although they will not be graded. These written assignments are intended to allow students to self-evaluate their level of preparedness and mastery of the course material, and students are encouraged to seek assistance from the instructor, teaching assistants, or each other in understanding and completing written assignments.

Readings will be assigned regularly, and students are expected to complete these readings in a timely fashion (no later than one week after they are assigned). Readings are selected to improve student understanding of the course material and/or present auxiliary material that the instructor believes is relevant and important. Material from readings may appear directly or indirectly in assignments, on quizzes, or on exams.

7.3 Tests and Quizzes

Quizzes may be introduced at any time by the instructor, covering any material previously covered in lectures, readings, or written homework assignments. These quizzes may or may not be announced in advance. (In particular, "pop quizzes" may be utilized to evaluate student attendance, engagement, and present understanding of course material.) There will be quizzes for each major lecture module and each lab activity.

There will be one midterm and one final exam. The midterm exam will cover all material presented in the course to date, including: lectures, labs, programming assignments, written homework assignments, and assigned readings. The (cumulative) final exam will cover all material covered in the course for the duration of the semester, including: lectures, labs, programming assignments, written homework assignments, and assigned readings.

7.4 Submission Policy

Programming assignments will be assigned with a deadline. All assignments are to be submitted by this deadline. In the event of any ambiguity in the deadline, times are assumed to be in the *current local time zone of the University*. Penalties for missing this deadline are as follows.

- Projects submitted before the deadline will incur no penalty.

- Projects submitted after the deadline, but within 24 hours of the deadline (excluding Saturday, Sunday, and University holidays) will incur a 20% penalty.

- Projects submitted more than 24 hours after the deadline as described above will not be accepted and will receive no credit.

Neither the instructor nor the teaching assistants will provide assistance for programming assignments after the assigned deadline.

Lab activities other than Lab 01 are due on the Monday following their assignment at 23:59. Lab 01 is due at 23:59 on the second Friday of classes (due to add/drop). No late submissions for lab activities will be accepted.

7.5 Programming Assignment Re-grading Policy

If you believe that a programming assignment has been graded incorrectly, you may submit it for re-grading. A request for a re-grade must be submitted within one calendar week of receiving the grade for a project, and must include:

- The original score achieved on the assignment

- A description of the specific error in grading that is being contested

- Relevant code demonstrating the submitted code's correctness or the grading script's incorrectness, if available

Re-grading of programming assignments is intended *only* to address errors in grading. No grades will be improved for any other reason, although they may be reduced; in particular, note that *your grade on any part of the assignment, not just the portion being re-graded, may be reduced* if re-grading discovers additional errors. This includes automated evaluations that passed because they did not trigger bugs that were discovered in manual evaluation for the re-grade, or bugs that show up only intermittently that happen to be encountered on the re-grade.

7.6 Exam and Quiz Re-grading Policy

If you believe that an exam or quiz has been graded incorrectly, you may submit it for re-grading. A request for a re-grade must be submitted within *one calendar week* of the exam or quiz being returned to you, must be submitted in writing (email is acceptable) to the instructor, and must include:

- The original, unmodified, exam or quiz answer

- A clear statement of the error

Re-grading of exams and quizzes is intended *only* to address errors in grading. No grades will be improved for any other reason, although they may be reduced if *errors are found in any portion of the assignment, not just the portion being re-graded.* Using re-grading as a bargaining tool to increase your score is likely to result in a lower grade, as the exam will be scrutinized in detail for errors that may have been missed the first time.

7.7 Make-up Policy

No deadline extensions or make-up work will be permitted except for approved University absences. Please see the University attendance policy for more information.

No make-up exams will be given whatsoever except for **documented extreme circumstances**. *24 hours of advance notice via e-mail must be provided if at all possible* before missing an exam session. If advance notice is not possible, documentation supporting this must be provided. Absence from an exam session due to illness **must** be supported by a note from a physician specifying that the student was too ill and/or contagious to attend on the exam date.

You are responsible for remembering and attending exam sessions. Please use extra assistance to remind yourself if necessary.

7.8 Contingencies related to COVID-19

There is a non-trivial chance that University operations will be disrupted in some way during this semester due to the ongoing spread of COVID-19. In the event that the execution of this course is disrupted, course staff will endeavor to provide continuity as best as possible. Unless circumstances otherwise dictate, lectures will move to online, live broadcast via a service such as YouTube Live, labs and office hours will move to a service such as Zoom, and the established course schedule will be maintained as far as is practical. A new syllabus will be issued only if required by the University, or if there are significant changes to the course material or grading policy.

If a student is required to miss class for COVID-related reasons that could not have been reasonably foreseen (*e.g.*, quarantine due to inadvertent exposure to the virus or contraction of the virus), reasonable accommodations will be made. Students who are quarantined but otherwise capable of continuing with course work are expected to view lectures online, attend virtual office hours, turn in assignments, and keep up with the course as much as is practical. The course is structured to allow this. Students who are too ill to pursue their course work will be accommodated as they would be for any other illness.

8 Course Schedule

The course schedule, including exam dates and times, is provided here for convenience. Note that course staff do not schedule many of the following items, including final exam time and location. *You are responsible for verifying your final exam time and location on HUB.* Inclement weather, local emergencies, unsafe building and/or campus environments, or other circumstances may cause the University to change this schedule. Course progress and pedagogical concerns may cause rescheduling of lectures, exams, and activities, or changes to required readings. You will be notified via Piazza or UB email of changes within the control of course staff. Course staff will attempt to keep you appraised of changes outside of staff control, but you are responsible for monitoring University communications to this effect.

You are responsible for monitoring any changes to this schedule, according to communications from course staff or the University. Failure to be aware of schedule changes is not sufficient reason for extended deadlines, make-up exams, or other accommodations.

Date	Description	Reading
2021-08-30	Introduction to CSE 220 and C	K&R Introduction, Ch. 1
2021-09-01	Variables, Strings, and Loops	K&R Ch. 1: 1.9, 1.10; Ch. 2: Intro, 2.1–2.4

Date	Topic	Reading
2021-09-03	Conditionals and Control Flow	K&R Ch. 2: 2.6; Ch. 3: 3.1 - 3.7
2021-09-08	Memory and Pointers	K&R Ch. 5: Intro, 5.1–5.4
2021-09-10	Programming Practices	
2021-09-10	**Programming Assignment 0 is due 23:59**	
2021-09-13	A Tour of Computer Systems	CS:APP Ch. 1: 1.1-1.7
2021-09-15	Memory Allocation	K&R Ch. 2: 2.7
2021-09-17	To Be Determined	
2021-09-20	Integers and Integer Representation	CS:APP Ch. 2: Intro, 2.1 through 2.1.3, 2.2
2021-09-22	(continued)	(ditto)
2021-09-24	Alignment, Padding, and Packing	K&R Ch. 5: Intro, 5.1–5.7; Chapter 6: Intro, 6.1–6.7; CS:APP Ch. 3: 3.8.1–3.8.3, 3.9.1, 3.9.3
2021-09-24	**Programming Assignment 1 is due 23:59**	
2021-09-27	Alignment, Padding, and Packing (continued)	(ditto)
2021-09-29	Floating Point Numbers	CS:APP Ch. 2: 2.4 Intro, 2.4.1–2.4.3, 2.4.6, 2.5
2021-10-01	(continued)	(ditto)
2021-10-04	Bitwise Operations	K&R Ch. 2: 2.9; Appendix A: A7.4.6, A7.8, A7.11-A7.13; CS:APP Ch. 2: 2.1.6 and 2.1.7
2021-10-06	Process Anatomy	K&R Ch. 4; CS:APP Ch. 3: 3.1, 3.7
2021-10-08	(continued)	(ditto)
2021-10-08	**Programming Assignment 2 is due 23:59**	
2021-10-11	Midterm Review	
2021-10-13	Midterm Exam	
2021-10-15	The Compiler and Toolchain	CS:APP Ch. 1: 1.1–1.4
2021-10-18	(continued)	(ditto)
2021-10-20	To Be Determined	
2021-10-22	Compiler Optimization	CS:APP Ch. 5: Intro, 5.1–5.6
2021-10-25	(continued)	(ditto)
2021-10-27	Dynamic Memory Allocation	
2021-10-29	Virtual Memory	CS:APP Ch. 1: 1.7.3; Ch. 9: Intro, 9.1–9.4
2021-10-29	**Programming Assignment 3 is due 23:59**	
2021-11-01	Virtual Memory (continued)	(ditto)
2021-11-03	Caching and Locality	CS:APP Ch. 6: Intro, 6.1–6.3, 6.5–6.7
2021-11-05	To Be Determined	
2021-11-08	Processes, Threads, and Concurrency	CS:APP Ch. 8: 8.2; Ch. 12: Intro, 12.1, 12.3
2021-11-10	(continued)	(ditto)
2021-11-12	Races and Synchronization	CS:APP Ch. 12: 12.4–12.7
2021-11-15	(continued)	(ditto)
2021-11-17	POSIX Threads and Synchronization	CS:APP Ch. 12: 12.3, 12.5–12.7
2021-11-19	(continued)	(ditto)
2021-11-19	**Programming Assignment 4 is due 23:59**	
2021-11-22	To Be Determined	

2021-11-24– 2021-11-28	Fall Recess	
2021-11-29	The Kernel and User Mode	CS:APP Ch. 8: 8.1, 8.2
2021-12-01	Input and Output	CS:APP Ch. 10: 10.1–10.4, 10.10–10.12
2021-12-03	Advanced Topics	
2021-12-06	Advanced Topics	
2021-12-08	Advanced Topics	
2021-12-10	Final Review	
2021-12-10	Programming Assignment 5 is due 23:59	
2021-12-16	Final Examination	

9 Grading Policy

No "I" (Incomplete) grades will be given for this course except for **documented extreme circumstances** or situations required by University policy. *Failure to complete work on time does not constitute an extreme circumstance.*

Grades will not be changed at the end of the semester for any reason other than a documented error in grading according to the policies outlined in Section 7.5 and Section 7.6. No grade negotiation will be permitted. In particular, no grades will be changed to preserve scholarships, fellowships, University positions, immigration status, internship or job offers, or any other outside factor. Grades reflect student performance and mastery of course material.

The credit breakdown for the course will be as follows:

Course Requirement	Course Grade	Percent
Less than 100% on Academic Integrity quiz	F	N/A
Less than 100% on Lab 01	F	N/A
Programming Assignment 0		5%
Programming Assignment 1		5%
Programming Assignment 2		5%
Programming Assignment 3		5%
Programming Assignment 4		10%
Programming Assignment 5		5%
Lab Activities		15%
Quizzes (Lab, Lecture, other)		20%
Midterm Exam		15%
Final Exam		20%

Note that this adds to 105%. This means that students who do all of their assigned, graded work have the opportunity to receive up to 5% extra credit in the course.

Failure to submit the Academic Integrity quiz with complete correctness (100% credit) or failure to submit Lab 01 with complete correctness (100% credit) will result in failure of the course. You may submit both of these assignments as many times as required to achieve complete correctness.

Final grades will be assigned from the above percentages as follows, although individual component scores may be adjusted or a curve of the instructor's choice may be applied if the instructor deems it warranted. Lower percentages are inclusive, upper percentages (excepting 100%) are not; that is, a 90.0% would be an A-, not a B+.

A	95-+ %	C+	77-80%
A-	90-95%	C	73-77 %
B+	87-90%	C-	70-73 %
B	83-87%	D+	67-70%
B-	80-83%	D	63-67%
		F	0-63%

10 Classroom Expectations

Students are expected to behave in a way that is respectful to their fellow students and the course staff. The University at Buffalo has a list of behavioral expectations that includes:

- Attending lectures and paying attention. Students should not ask an instructor in class to go over material they missed by skipping a class or not concentrating. If lectures are watched asynchronously, students should watch them on the day they are given or as close as reasonably possible.

- Not coming to class late or leaving early. If a student has to enter a class late, he or she should do so quietly and should not disrupt the class by walking between the class and the instructor. Students should not leave class unless it is an absolute necessity.

- Not talking with other classmates while the instructor or another student is speaking. If a student has a question or comment, he or she should raise a hand, rather than starting a conversation about it with a neighbor.

- Showing respect and concern for others by not monopolizing class discussion. Students must allow others time to give their input and ask questions. Students should not stray from the topic of class discussion.

- Not eating and drinking during class time.

- Turning off electronic devices including cell phones, pagers, and watch beepers.

- 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, watching videos, and exploring the Internet are discouraged.

- Not packing bookbags or backpacks to leave until the instructor has dismissed class.

11 Academic Integrity

Students will abide by the CSE Academic Integrity Policy, the University Academic Integrity Policy, and the Undergraduate or Graduate amendments thereof, as appropriate.

The Academic Integrity policy for this course, and my other courses, can be found on my web site, under Policies. You should read it for additional information and clarifications not found here.

All resources used in completing assignments for this class *must be given appropriate attribution*, and the *only resources allowed for the completion of programming assignments, quizzes, or exams without specific permission* are as follows.

- The required course textbook *Computer Systems: A Programmer's Perspective*, Third Edition, by Bryant and O'Hallaron

- The required course textbook *The C Programming Language,* by Kernighan and Ritchie

- Lecture material from this course

- Required or recommended readings from lecture material

- Man and info pages from the course virtual machine

In particular, Stack Exchange, code from other students in the course or students who have completed this course or related courses at other universities in previous semesters, GitHub repositories, code or algorithms from other web sites or books, and other resources are not allowed without explicit permission from the instructor.

If there is any question about whether a resource is acceptable for use in completing a course assignment, students are encouraged to ask the instructor or a TA *before* making use of it. Asking about a resource is **not** a violation of academic integrity, even if the resource is not allowed for the course.

Quizzes and exams may have further restrictions on allowable resources; for example, a student's own work from previous assignments may not be an allowable resource on an exam.

Violation of these policies will result in a failing grade for the course and referral upward for additional sanctions according to University policy.

Sharing of course materials *after* completing this course is also a violation of the academic integrity policy for this course.

11.1 Sharing CSE 220 Materials After Completion

Materials used in CSE 220, including project handouts *and your own project implementations*, remain a part of this course after you complete it. Sharing those materials after completing CSE 220 may still constitute an academic integrity violation. Academic integrity proceedings may be started **even after you have passed this course**.

In particular, be aware that if you post your CSE 220 project material in a public place after completing CSE 220, **you may be subject to academic integrity proceedings**. This can result in a **retroactive failure** in this course, which will render you ineligible for enrollment in any courses that require CSE 220 or for graduation until you retake CSE 220.

11.2 Academic Integrity Amnesty

A student who has committed a violation of this academic integrity policy may receive limited amnesty for the violation by *notifying me, in writing,* of the violation **before I have begun to assess the violating assignment.** This notification must include the student's name, person number, UBITname, and state the assignment in question and the nature of the violation. Upon submitting such a statement, the student will receive no credit for the violating assignment, but *no further sanctions will be taken, and the violation will not be reported.* Once I have begun assessing the assignment in question, no such statements will be permitted. Since it may not be obvious to students when assessment begins, such statements should be submitted as soon as possible after the violation occurs. While assessment may begin at any time, in general I will not look at student submissions until a project deadline has passed.

See my online Academic Integrity Policy for an example scenario and more information.

12 Program Outcomes and Competencies

This course is designed to fulfill a specific role in your education here at the University at Buffalo, as part of a larger curriculum to provide you with the foundations to propel you into a successful career as a computer scientist or engineer. As a 200-level course, CSE 220 provides less structure and requires more personal responsibility than 100-level courses, but in turn provides more structure and guidance than 300- and 400-level courses. You will learn not only technical details and skills, but also apply personal time management and study skills in ways that may be new to you.

12.1 Learning Objectives

Students who pass CSE 220 with an A or B will have demonstrated, through examination or practical implementation, substantial mastery of the topics listed here. Students who pass CSE 220 with a passing but lesser grade will have demonstrated competence appropriate for moving on to future courses at UB in the topics listed here.

Numeric Representations: Students will translate decimal integers to and from their binary representation. They will identify and manipulate the three parts of an IEEE-754 floating point number (sign, significand, and exponent), as well as convert to and from decimal in the case of terminating binary fractional numbers of a modest number of digits.

Memory Organization: Students demonstrate an ability to manipulate user-space memory in a Unix process via address calculation and direct manipulation by pointer casting. They will show the ability to identify the relevant sections of a statically-linked Unix process (text, data, BSS, stack, and heap).

Computer Architecture: Students identify the various logical elements of a modern computer, including CPU, cache, main memory, secondary storage, and data and address busses. They demonstrate an understanding of the various levels of memory in the storage hierarchy, including CPU registers, various levels of cache, main memory, and secondary local or remote storage. They will demonstrate the ability to calculate memory alignments and identify certain caching effects.

Compilers and Compiler Optimization: Students will show the ability to identify the typical stages of C compilation (preprocessing, compilation, assembly, and compile-time linking) and the duties of each stage. They will use the C compiler to produce executables from multiple compilation units. They will demonstrate the ability to identify simple compiler optimizations, as well as some circumstances where compiler optimization is blocked by specific program construction or language semantics.

Operating Systems: Students will establish their understanding of the basic duties of an operating system, including: the process abstraction, isolation of user-space processes, management and sharing of hardware resources, memory protection and virtual memory management, and the system call interface.

Concurrency: Students will implement simple multi-threaded data structures. They will demonstrate the ability to use mutual exclusion to protect critical sections and enable the safe sharing of memory between threads in an application.

12.2 ABET Accreditation Outcomes

This course is required in both the BS Computer Engineering program, accredited by the Engineering Accreditation Commission (EAC) of ABET, and the BS Computer Science program, accredited by the Computing Accreditation Commission (CAC) of ABET.

The course introduces students to the following CAC student outcomes, for which graduating students must demonstrate:

(CAC-1) Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions.

(CAC-2) Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

(CAC-6) Apply computer science theory and software development fundamentals to produce computing-based solutions.

The course introduces students to the following EAC student outcomes, for which graduating students must demonstrate:

(EAC-1) an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics

(EAC-7) an ability to acquire and apply new knowledge as needed, using appropriate learning strategies

Student outcomes will be evaluated as follows.

CAC 1	CAC 2	CAC 6	EAC 1	EAC 7	Assessment Types
✓	✓	✓	✓	✓	Programming Assignments
✓	✓	✓	✓	✓	Lab Activities
✓			✓		Exams

13 Accessibility Resources

From the UB Reasonable Accomodation Policy:

> The University at Buffalo is committed to providing equal access to individuals with disabilities, including physical access to programs and reasonable accommodations for members of the university community.

The UB Accessibility Resources Office provides assistance for students who require reasonable accommodations due to disability. They may be found at 60 Capen Hall or contacted by phone at 716-645-2608. Students must register with their office to receive accommodations for physical or learning disabilities.

14 Critical Campus Resources

Sexual Violence   UB is committed to providing a safe learning environment free of all forms of discrimination and sexual harassment, including sexual assault, domestic and dating violence and stalking. If you have experienced gender-based violence (intimate partner violence, attempted or completed sexual assault, harassment, coercion, stalking, etc.), UB has resources to help. This includes

academic accommodations, health and counseling services, housing accommodations, helping with legal protective orders, and assistance with reporting the incident to police or other UB officials if you so choose. Please contact UB's Title IX Coordinator at 716-645-2266 for more information. For confidential assistance, you may also contact a Crisis Services Campus Advocate at 716-796-4399.

Mental Health   As a student you may experience a range of issues that can cause barriers to learning or reduce your ability to participate in daily activities. These might include strained relationships, anxiety, high levels of stress, alcohol/drug problems, feeling down, health concerns, or unwanted sexual experiences. Counseling, Health Services, and Health Promotion are here to help with these or other issues you may experience. You can learn more about these programs and services by contacting:

- Counseling Services:
    - 120 Richmond Quad (North Campus), 716-645-2720
    - 202 Michael Hall (South Campus), 716-829-5800

- Health Services:
    - Michael Hall (South Campus), 716-829-3316

- Health Promotion:
    - 114 Student Union (North Campus), 716-645-2837

Acknowledgments

Some language in this syllabus is drawn from University policies (as noted), the UB Course Syllabi Requirements document, department guidelines, and other University resources. Some language and structure in this syllabus is drawn from Steve Ko's CSE 486/586 syllabus from Spring 2017 and from Matthew Hertz's CSE 115/503 syllabus from Spring 2019. Many improvements to this syllabus were made by Karthik Dantu in Fall 2019.