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.

The syllabus is subject to change based on the needs of the course and will be communicated with you as appropriate.

Instructors

Name	Email	Office	Office Hours
Dr. Eric Mikida	epmikida@buffalo.edu	208 Capen Hall	TBD
Dr. Shamsad Parvin	shamsadp@buffalo.edu	313 Davis Hall	TBD

Course Information

CSE 4/587 Data Intensive Computing – 3 credits

Course Description

Data-intensive computing deals with storage models, application architectures, middleware, and programming models and tools for large-scale data analytics. In particular we study approaches that address challenges in managing and utilizing ultra-scale data and the methods for transforming voluminous data sets (big data) into discoveries and intelligence for human understanding and decision making. Topics include: storage requirements of big data, organization of big data repositories such as Google File System (GFS) semantic organization of data, data-intensive programming models such as MapReduce, fault-tolerance, privacy, security and performance, services-based cloud computing middleware, intelligence discovery methods, and scalable analytics and visualization. This course has three majors goals: (i) understand data-intensive computing, (ii) study, design and develop solutions using data-intensive computing models such as MapReduce and (iii) focus on methods for scalability using the cloud computing infrastructures such as Google App Engine (GAE), Amazon Elastic Compute Cloud (EC2), and Windows Azure. On completion of this course students will be able to analyze, design, and implement effective solutions for data-intensive applications with very large scale data sets.

Course Learning Outcomes

1. On completion of this course students will be able to analyze, design, and implement effective solutions for data-intensive applications with very large scale data sets. More specifically a student will be able to:
2. Recognize a data-intensive problem.
3. Assess the scale of data and requirements.
4. Retrieve data using appropriate methods.
5. Describe the data layout and define the data repository format (Ex: store).
6. Decide the algorithms (Ex: MapReduce) and programming models (Ex: Bayesian).
7. Define application-specific algorithms and analytics (Ex: network analysis).
8. Design the data-intensive program solution and system configuration.
9. Implement the data-intensive solution and test the solution.
10. Write a report summarizing the solution and results.
11. Incorporate services from cloud computing platforms.
12. Study the foundational concepts enabling cloud computing: services-based interface, programmatic consumption of services, virtualization, PKI-based security, large-scale storage, load-balancing, machine images and on-demand services.
13. Formulate data-intensive visualization solutions for presenting the results.

Prerequisites

CSE 250 and approved Computer Science, Computer Engineering, Bioinformatics/CS Majors only. Departmental senior standing recommended. Students must complete a mandatory advisement session with their faculty advisor.

Textbooks

There are two recommended texts:

1. Doing Data Science: Straight Talk from the Frontline, 1st Edition Author(s): Cathy O'Neil and Rachel Schutt ISBN: 978-1449358655 Publisher: O'Reilly Media
2. Joel Grus. Data Science from Scratch: First Principles with Python 1st Edition, 978-1491901427, O'Reilly media. 2015.

We will be using many other references and online sources throughout the semester. Links will be posted on the course website.

Computing Resources

You will be using various free on-line tools for this course – links will be posted on the course website. Course-related communications should be via the Piazza forum linked from the course website. Piazza posts can be either public to the class or private to instructors. Any email communications must come from your UB email account and include [CSE 487] or [CSE 587] in the subject line. All communications with course staff are expected to be professional. Graded work will be both submitted and returned via UBLearns.

Attendance Policy

The conceptual and theoretical course content will be delivered primarily through the in-person lectures. You are expected to attend lectures and take your own notes to prepare for later assessments. If you are out of class for an extended period of time because of sickness, notify your instructor as soon as possible. If you miss a significant portion of the semester it is recommended that you resign from the course.

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 them to be necessary. Any changes will be communicated to the class in writing via e-mail to each student's UB e-mail account.

Requirement	Weight	Details
Homework	10%	Two graded homework assignments on topics not covered by the project.
Course Project	40%	Three part project over the duration of the semester to be completed in teams.
Midterm Exams	20%	Two in-class midterm exams during the course of the semester.
Final Exam	30%	Cumulative final exam during the final exam period covering all class material.

Overall course grade

Percentage	Letter grade	Percentage	Letter grade
95 - 100	A	70-74	C+
90 - 94	A-	65-69	C
85 - 89	B+	60-64	C-
80-84	B	55-59	D
75-79	B-	0-54	F

Regrading

Any questions about the grading of a piece of work ***must be raised within one week of the date that the graded work was returned to you.***

Incomplete (I) grades

A grade of incomplete ("I") indicates that additional coursework is required to fulfill the requirements of a given course. Students may only be given an "I" grade if they have a passing average in coursework that has been completed and have well-defined parameters to complete the course requirements that could result in a grade better than the default grade. An "I" grade may not be assigned to a student who did not attend the course.

Prior to the end of the semester, students must initiate the request for an "I" grade and receive the instructor's approval. Assignment of an "I" grade is at the discretion of the instructor.

The last day to resign the course is **Thursday, March 2, 2023**.

Academic Integrity

Academic integrity is a fundamental university value. Through the honest completion of academic work, students sustain the integrity of the university while facilitating the university's imperative for the transmission of knowledge and culture based upon the generation of new and innovative ideas. Please refer to the university Undergraduate Academic Integrity Policy (*https://catalog.buffalo.edu/policies/academic_integrity_2019-20.html*) for additional information.

As an engineer or computer scientist, you have special ethical obligations. As per the NSPE Code of Ethics, "engineers shall avoid deceptive acts" and "shall conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession (*https://www.nspe.org/resources/ethics/code-ethics*). Similar sentiments of honesty, integrity, fairness, and responsibility are fundamental to the ACM Code of Ethics (*https://www.acm.org/code-of-ethics*).

A violation in this class generally results in an F for the entire course. The Computer Science and Engineering department's policy on academic integrity can be found here:

https://engineering.buffalo.edu/computer-science-engineering/information-for-students/undergraduate-program/cse-undergraduate-academic-policies/cse-academic-integrity-policy.html

What Constitutes a Violation of Academic Integrity?

These bullets should be obvious things not to do (but commonly occur):

- Turning in your friend's code/write-up (obvious).
- Turning in solutions you found on Google with all the variable names changed (should be obvious). This is a copyright violation, in addition to an AI violation.
- Turning in solutions you found on Google with all the variable names changed and 2 lines added (should be obvious). This is also a copyright violation.
- Paying someone to do your work. You may as well not submit the work since you will fail the exams and the course.
- Posting to forums asking someone to solve the problem. Note: Aggregating every [stack overflow answer|result from google|other source] because you "understand it" will likely result in full credit on assignments (if you aren't caught) and then failure on every exam. Exams don't test if you know how to use Google, but rather test your understanding (i.e., can you understand the problems to arrive at a solution on your own). Also, other students are likely doing the same thing and then you will be wondering why 10 people that you don't know have your solution.

Other violations that may not be as obvious:

- Working with a tutor who solves the assignment with you. If you have a tutor, please contact me so that I may discuss with them what help is allowed.
- Sending your code to a friend to help them. If another student uses/submits your code, you are also liable and will be punished.
- Joining a chatroom for the course where someone posts their code once they finish, with the honor code that everyone needs to change it in order to use it.
- Reading your friend's code the night before it is due because you just need one more line to get everything working. It will most likely influence you directly or subconsciously to solve the problem identically, and your friend will also end up in trouble.

What Collaboration is Allowed?
Assignments in this course should be solved individually with only assistance from course staff and allowed resources. You may discuss and help one another with technical issues, such as how to get your compiler running, etc.

There is a gray area when it comes to discussing the problems with your peers and I do encourage you to work with one another to solve problems. That is the best way to learn and overcome obstacles. At the same time you need to be sure you do not overstep and not plagiarize. Talking out how you eventually reached the solution from a high level is okay:

> "I used a stack to store the data and then looked for the value to return."

but explaining every step in detail/pseudocode is not okay:

> "I copied the file tutorial into my code at the start of the function, then created a stack and pushed all of the data onto the stack, and finished by popping the elements until the value is found and use a return statement."

The first example is OK but the second is basically a summary of your code and is not acceptable, and remember that you shouldn't be showing any code at all for how to do any of it. Regardless of where you are working, you must always follow this rule: **Never come away from discussions with your peers with any written work, either typed or photographed, and especially do not share or allow viewing of your written code.**

What Resources are Allowed?
With all of this said, please feel free to use any [files|examples|tutorials] that we provide directly in your code (with proper attribution). Feel free to directly use anything from lectures or recitations. You will never be penalized for doing so, but should always provide attribution/citation for where you retrieved code from. Just remember, if you are citing an algorithm that is not provided by us, then you are probably overstepping.

More explicitly, you may use any of the following resources (with proper citation/attribution in your code):
- Any example files posted on the course webpage (from lecture or recitation).
- Any code that the instructor provides.
- Any code that the TAs provide.

Omitting citation/attribution will result in an AI violation (and lawsuits later in life at your job). This is true even if you are using resources provided.

Amnesty Policy

We understand that students are under a lot of pressure and people make mistakes. If you have concerns that you may have violated academic integrity on a particular assignment, and would like to withdraw the assignment, you may do so by sending me an email **BEFORE THE VIOLATION IS DISCOVERED BY ME.** The email should take the following format:

> Dear Dr. Mikida/Dr. Parvin,
>
> I wish to inform you that on assignment X, the work I submitted was not entirely my own. I would like to withdraw my submission from consideration to preserve academic integrity.
>
> J.Q. Student
> Person #12345678
> UBIT: jqstuden

When we receive this email, student J would receive a 0 on assignment X, but would not receive an F for the course, and would not be reported to the office of academic integrity.

Critical Campus Resources

Accessibility Resources

If you have any disability which requires reasonable accommodations to enable you to participate in this course, please contact the Office of Accessibility Resources in 60 Capen Hall, 716-645-2608 and also the instructor of this course during the first week of class. The office will provide you with information and review appropriate arrangements for reasonable accommodations, which can be found on the web at: http://www.buffalo.edu/studentlife/who-we-are/departments/accessibility.html.

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:

- 4350 Maple Rd, Amherst, NY 14226, 716-829-3316

Health Promotion:

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

Diversity

The UB School of Engineering and Applied Sciences considers the diversity of its students, faculty, and staff to be a strength, critical to our success. We are committed to providing a safe space and a culture of mutual respect and inclusiveness for all. We believe a community of faculty, students, and staff who bring diverse life experiences and perspectives leads to a superior working environment, and we welcome differences in race, ethnicity, gender, age, religion, language, intellectual and physical ability, sexual orientation, gender identity, socioeconomic status, and veteran status.