جامعة نيويورك أبوظبي
NYU | ABU DHABI

CS-UH 3260
Special Topics in Computer Science
Big Data Systems
Spring, 2023

Credits: 4

Prerequisites: CS-UH 1052 (Algorithms)

Recommended Prerequisites: CS-UH 2214 (Database Systems) or CS-UH 2012 (Software Engineering) or CSCI-UA 479 (Data Management and Analysis) or CSCI-UA 60 (Database Design and Implementation)

Corequisites: NONE

Faculty Details	**Professor**	**Teaching Assistant**
Name	Djellel Difallah	Sara Mumtaz
Email	djellel@nyu.edu	sm11206@nyu.edu
Telephone	+917 2 628 7380	TBD
Workspace	A1-185	A2-186F
Office Hours	By Appointment	Wed 2.30 PM -3.30 PM

Course Details	Day/Time	Location
Lecture	Tue/Thu 3:35 PM - 4:50 PM	Lecture room
Mid Term Exam	Last session before Spring Break	Lecture room
Final Exam	Finals week	TBD

This course counts toward the following NYUAD degree requirements:

- Majors > Computer Science > Computer Science Elective Course

Course Description

Businesses, governments, individuals, scientific tools, and smart devices continuously create massive amounts of data at an unprecedented scale. Mining or even querying such datasets have become increasingly difficult. Standard database systems were not designed to deal with the size and dynamic format of datasets. This course focuses on conceptual and architectural issues related to designing and deploying data management systems in a "big data" context (Large data *volume, velocity, and variety*).

The course will first review distributed transaction processing techniques, classical parallel database systems, and ACID-style semantics in shared-nothing architectures. Then, we will delve into various distributed data processing techniques, with an emphasis on solutions based on clusters of commodity machines and handling embarrassingly parallel computing problems. Particularly, we will cover distributed storage systems (such as Google File system, HDFS), wide-area hash-tables (like Cassandra), data-intensive computing platforms (Hadoop, Spark), and NoSQL systems. Hands-on programming exercises and deployments of such platforms will be a central part of the course.

Course Learning Outcomes and Link to Program Learning Outcomes (PLOs)

Course Learning Outcomes	Linked to X Major PLOs[1]	Level of Contribution to PLO
1. Develop an understanding of big data challenges and solutions. The student will be able to make big data design choices for specific workloads	PLO 1	High
2. Master programming techniques with big data platforms. The student will be able to write Map-Reduce and Spark scripts to perform data manipulations on large-scale datasets.	PLO 1 PLO 2 PLO 3 PLO 6	High Medium High Low
3. Build a hands-on experience deploying big data systems. This includes generic installation and configuration of such systems on a server or a cluster.	PLO 2 PLO 3 PLO 6	High Low Low
4. Understand the theoretical aspects of big data solutions, including challenges related to distributed computing, storage, and their current solutions.	PLO 1 PLO 2 PLO 5	High High Low

[1] See Appendix 1

Teaching Methodologies

- **Direct Instruction in the form of Lectures, Questions, and in-class Discussions.** The course will explain, through several scenarios, the data management and processing challenges where traditional RDBMS technology has proven inadequate. We will then introduce current solutions, their architecture, and (when applicable) algorithms describing the inner workings of the system.
- **Directed Self-Learning by Reading.** Lecture slides are made available to the students. The students will be provided with additional resources, such as research papers in computer science.
- **Programming and writing.** Regular programming assignments will be assigned to help the students get acquainted with the tools and techniques as we progress through the course. For each assignment, the deliverable will be in the form of a short report and programs or scripts.
- **Exam assessments**. A midterm and final assessments will test knowledge of theoretical concepts learned in the course

Graded Activities

Activity Detail	Grade Percentage	Submission Date/Week	Linked to Course Learning Outcome(s)
Assignments (6)	60% (10% each)	Biweekly (or as per the schedule)	1, 2, 3, 4
Midterm Exam	20%	Week 7	1, 2, 4
Final Exam	20%	finals week	1, 2, 4

Grade Distribution: Syllabi for courses that use percentage grades on individual assignments should ordinarily contain a rubric which converts percentage grades to letter grades, both for individual assignments and for the course as a whole. Students need to obtain a grade of C or better to count the course towards their major/minor/core/prerequisite requirement.

A	A-	B+	B	B-	C+	C	C-	D+	D	F
[95-100]	[90-95)	[87-90)	[83-87)	[80-83)	[77-80)	[73-77)	[70-73)	[67-70)	[63-67)	[0-63)

Course Schedule:
Below is a **provisional schedule** of the topics to be covered in this course. Dates are subject to confirmation and may change. Please make sure to go over the readings before coming to class.

Week	Session	Topic	Reading	Other
1	Session 1	*Introduction: Big Data Challenges + Relational Databases*	[MMD] ch.1	*Course Survey and SQL primer*
	Session 2	Continued lecture + lab (Infrastructure Setup)		
2	Session 1	*Review of Database Concepts: Storage, Indexing, Transactions, Parallel DBMSs, CAP*	[SQL] Sections 1-5 [CAP]	*Assignments 1 Published*
	Session 2	Continued lecture + lab (DB Install and SQL)		
3	Session 1	*Overview on Peer-to-Peer Systems*	[CHR] Sections 1-4, 6-7	
	Session 2	Continued lecture + lab		*Assignments 1 Due*
4	Session 1	*Data Analytics with Column Stores*	[CST]	*Assignments 2 Published*
	Session 2	Continued lecture + lab (MonetDB)		
5	Session 1	*NOSQL: Document Stores, Graph Databases, Array Data Management*	[DYN]	
	Session 2	Continued lecture + lab (MongoDB, Neo4J)		*Assignments 2 Due*
6	Session 1	*Key-Value Stores: Memcached, Cassandra*	[CAS]	*Assignments 3 Published*
	Session 2	Continued lecture + lab (Memcahed, Redis)		
7	Session 1	*Distributed Graph Processing: Pregel, Apache Giraph*	[DGP] Sections 8.1-3 [PRG]	
	Session 2	**Midterm Exam**		*Assignments 3 Due*
Break				

8	Session 1	*Distributed File Systems: Google File System + HDFS*	[DTP] Ch.2.5 [MMD] Ch.2 [HDG] Ch.3	*Assignments 4 Published*
	Session 2	Continued lecture + lab (HDFS / Parquet)		
9	Session 1	*Introduction to MapReduce and its Design Patterns*	[DTP] Ch.1, Ch.2, Ch.6 [MAP]	
	Session 2	Continued lecture + lab (Word count, Terasort)		
10	Session 1	*Apache Hadoop + Config (Zookeeper) + Scheduling (YARN)*	[HDG] Ch.4	
	Session 2	Continued lecture + lab (Setting up Queues)		*Assignments 4 Due*
11	Session 1	*Apache Spark*	[SPK] Ch.1, Ch.3	*Assignments 5 Published*
	Session 2	Continued lecture + lab (Spark)		
12	Session 1	*Stream Processing (Spark Streaming) and Event-streaming (Kafka)*	[MML] Ch.4 [SPK] Ch.8 [KFK] Ch.1	
	Session 2	Continued lecture + lab (Kafka)		*Assignments 5 Due*
13	Session 1	*Big Data Integration and Cleaning*	Lecture Notes or Guest Lecture (TBD)	*Assignments 6 Published*
	Session 2	Continued lecture + lab		
14	Session 1	*Big Data and Machine Learning: Snorkel*	[SNO]	
	Session 2	Course Review		*Assignments 6 Due*

Required Bookstore Texts

- There is no official required textbook; instead a collection of reference readings available online will be provided throughout the course, in addition to recommended books.

Provisional list of readings and references

- [MMD] Mining of Massive Datasets. (Leskovec, Rajaraman, Ullman) 3rd edition. Link
- [SQL] http://philip.greenspun.com/sql/index.html
- [CAP] Abadi, Daniel. "Consistency tradeoffs in modern distributed database system design: CAP is only part of the story." Computer 45.2 (2012): 37-42.

- [CHR] Stoica, Ion, et al. "Chord: A scalable peer-to-peer lookup service for internet applications." ACM SIGCOMM computer communication review 31.4 (2001): 149-160.
- [CST] Stonebraker, Mike, et al. "C-store: a column-oriented DBMS." Making Databases Work: the Pragmatic Wisdom of Michael Stonebraker. 2018. 491-518.
- [CAS] Apache Cassandra Talk (by Benoit Peroud). Slides
- [DYN] Sivasubramanian, Swaminathan. "Amazon DynamoDB: a seamlessly scalable non-relational database service." Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data. 2012.
- [DGP] https://stanford.edu/~rezab/classes/cme323/S15/notes/lec8.pdf
- [PRG] Malewicz, Grzegorz, et al. "Pregel: a system for large-scale graph processing." Proceedings of the 2010 ACM SIGMOD International Conference on Management of data. 2010.
- [DTP] Data-Intensive Text Processing with MapReduce. Jimmy Lin and Chris Dyer (2013). Link
- [HDG] Hadoop: The Definitive Guide. by Tom White (O'Reilly 4th Edition).
- [MAP] Dean, Jeffrey, and Sanjay Ghemawat. "MapReduce: simplified data processing on large clusters." *Communications of the ACM* 51.1 (2008): 107-113.
- [SPK] Learning Spark Lightning-Fast Data Analytics. by Damji, Wenig, Das & Lee. Link
- [KFK] Kafka: The Definitive Guide. by Narkhede, Shapira, Palino (O'Reilly 2017)
- [SNO] https://snorkel.ai/how-to-use-snorkel-to-build-ai-applications/
- *Additional readings from online resources and published papers could be assigned during the term of the course.*

Assignments and Hands-on Experience:

For the hands-on parts of the assignments, you will be writing scripts for distributed data processing. We will be using Python most of the time. However, due to the nature of such platforms, you will have to acquire practical hands-on experience with new languages, such as SQL or Scala. You will have the lecture notes, external resources, and my support to pick up the minimally required skills effectively. Therefore, I encourage you to work with your classmates to review the labs, and then complete your assignment independently.

Theoretical Assessments:

While this is not an algorithms class, you can expect to learn about (a) selected distributed systems algorithms, (b) new data structures, (c) data processing patterns to use under different workloads, and (d) software architecture principles behind big data systems.

Academic Policies

Attendance: Attendance for courses is mandatory. Every absence needs to be agreed upon with the professor prior to class.

Class Participation: Class participation is strongly encouraged as it helps to grasp and understand the material.

Collaboration: You are welcome to work with other students to resolve technical issues in homeworks, but your hand-ins must be entirely your own. Collaboration on exams is not allowed.

Lab sessions and assignments: You must have access to a computer on which you can install software, such as a virtual machine. You should bring your laptop to class when a lab session is announced.

Assignment Policy:

- Assignments will be released on NYU Brightspace with the due date specified.
- Assignments are handed in and returned on NYU Brightspace.
- You may discuss with your classmates about the assignments, but you need to organize and write your solutions by yourself.
- Appeals should be made within one week after the graded assignment is returned.
- Late submission: You can make up to 96 hours (4 days) of late submissions **cumulatively**.
    - There will be 20% off for every additional late 24 hours on subsequent assignments.

Integrity: At NYU Abu Dhabi, a commitment to excellence, fairness, honesty, and respect within and outside the classroom is essential to maintaining the integrity of our community. By accepting membership in this community, students, faculty, and staff take responsibility for demonstrating these values in their own conduct and for recognizing and supporting these values in others. In turn, these values create a campus climate that encourages the free exchange of ideas, promotes scholarly excellence through active and creative thought, and allows community members to achieve and be recognized for achieving their highest potential.

Students should be aware that engaging in behaviors that violate the standards of academic integrity will be subject to review and may face the imposition of penalties in accordance with the procedures set out in the NYUAD policy: https://students.nyuad.nyu.edu/campus-life/student-policies/community-standards-policies/academic-integrity/

NYU Moses Center for Student Accessibility

New York University is committed to providing equal educational opportunity and participation for students with disabilities. The center works with NYU students to determine appropriate and reasonable accommodations that support equal access to a world-class education. Confidentiality is of the utmost importance. Disability-related information is never disclosed without student permission. Find further information at:

https://www.nyu.edu/students/communities-and-groups/students-with-disabilities.html

Contact: mosescsd@nyu.edu

Appendix 1

Computer Science Major Program Learning Outcomes (PLOs)

PLO 1 Analyze a problem, and identify, define, and verify the appropriate computational tools required to solve it (Knowledge, Skill, Role in Context, Self-development).

PLO 2 Apply up-to-date computational tools necessary in a variety of computing practices (Knowledge, Skill, Autonomy & Responsibility, Self-development).

PLO 3 Implement algorithms as programs using modern computer languages (Knowledge, Skill).

PLO 4 Apply their mathematical knowledge to solve computational problems (Knowledge, Skill, Autonomy & Responsibility, Self-development).

PLO 5 Communicate computer science knowledge both orally and in writing (Skill, Autonomy & Responsibility, Role in Context).

PLO 6 Collaborate in teams (Skill, Autonomy & Responsibility, Role in Context).