CS6240: Large-Scale Parallel Data Processing

For all general course information such as credit hours, format, meeting times and location, please refer to the registrar system for the latest information.

Instructor Information: Dr. Mirek Riedewald
Office Hours, Email, TA: this information will be posted on Canvas

Following university policy, this course is currently scheduled as a regular in-person lecture on the Boston campus. Students are expected to attend all lectures in person and take the exam in the classroom in person.

Please be aware of the following policies:

- There are **no** deadline extensions or make-up assignments/exams, except if you have a major emergency. You must provide evidence to claim such an emergency and you must inform the instructor *as soon as possible*. The following are examples for situations that do *not* qualify as emergencies: (i) I have job/co-op/internship interviews. (ii) My other course has an exam. (iii) My other course has a major homework or project deadline.
- We understand that some weeks are busier than others, but that's how things will be in your future job as well. By announcing deadlines well in advance, we give you the opportunity to plan and schedule your work accordingly. Make sure you start early so that you have the flexibility for dealing with unexpected issues.
- **Honor Code**: All students must adhere to the Northeastern University honor code available on the Northeastern web site and the graduate student handbook.
    - Please note that you are *not* allowed to share homework solutions with others or copy anybody else's homework entirely or in parts. We will check for originality during the grading process.
    - If you use someone else's code, text, etc, you must clearly indicate the copied material and properly cite the source. This also applies to material that you "slightly" modify. If in doubt, cite it and briefly explain or highlight how you modified it.

Course Prerequisites and Description: See the official information in the course catalog.

Course Format & Methodology: This course runs for a total of 15 weeks and contains online content accessible through http://khoury.northeastern.edu/~mirek/teaching.htm and https://canvas.northeastern.edu/ Each week (or module) contains one or more lessons, which need to be completed by Sunday of the week *before* the module is discussed. Homework and project solutions will be managed in Gradescope. For source code management, the instructor will create GitHub Classroom repositories. **Please note that all due dates and times are specified according to the local Boston time (Eastern US time zone)**.

Recommended Textbook & Materials: There is *no required textbook* because the instructor provides textbook-like course material. To gain a deeper understanding of the material covered in this course, we recommend the following books (most should be available online for free for Northeastern University students from O'Reilly for Higher Education):

- *MapReduce Design Patterns* by Donald Miner and Adam Shook
- *Hadoop: The Definitive Guide* by Tom White
- *High Performance Spark* by Holden Karau and Rachel Warren
- *Spark: The Definitive Guide* by Bill Chambers and Matei Zaharia
- *Spark in Action* by Petar Zecevic and Marko Bonaci
- *Programming Elastic MapReduce* by Kevin Schmidt and Christopher Phillips

For some topics we will work with research papers or other online resources, e.g., the Hadoop and Spark API doc.

Course Outcomes: This course has the following main objectives and content:

- Get an overview of the big-data-processing landscape.
    - We will discuss some trends and challenges and briefly survey alternative approaches.
- Learn how to design distributed algorithms for processing big data, and how to implement them in Hadoop MapReduce and in Spark. While MapReduce or Spark might be replaced at some point by other systems, the algorithm design patterns taught in this course will remain relevant, because they are concerned with partitioning of a problem and assigning data to many machines.
    - We will cover a variety of fundamental problems and design patterns, including join computation, graph algorithms, information retrieval and data mining techniques, and analyze how they can be implemented in a scalable manner.
- Get hands-on practice writing code and running it on many processors.
    - We will work with Hadoop MapReduce and Spark.
    - We will use the Amazon Cloud to run the code but may work with a different provider if necessary. (Our goal is to provide a real-world commercial-cloud experience at minimal cost—ideally zero—for each student.) Details will be announced with the first homework assignment.
- Understand the system architecture and functionality below MapReduce and Spark.
    - We will discuss features and limitations of MapReduce and Spark.

Notice that we cannot cover all possible parallel-computation approaches. You are encouraged to explore other courses on related topics. Also note that new approaches for big-data processing keep appearing, often trying to address some weakness of existing ones. We will not be able to cover them at this point, but a solid understanding of parallel-data-processing principles will help you evaluate their tradeoffs—something the marketing people probably will not tell you about…

Participation and Engagement: Your presence in peer-to-peer activities serves as an indicator of your level of engagement and effort throughout the course. Frequent and varied (e.g., synchronous/asynchronous/face-to-face) opportunities to receive feedback, help, and clarification on course material from the instructor are provided throughout the term. The following activities count towards class participation:

1. Asking or answering questions in class.
2. Submitting solutions for in-class exercises when requested by the instructor.
3. Answering questions or posting relevant information in the discussion boards. Participation points are awarded based on quality and quantity of contributions.

Communication/Submission of Work: Make sure you receive course-related announcements the day they are made. Guidelines for completing and submitting each assignment are posted along with the assignment. Late and early homework submission policies will be announced with the individual assignments.

Course Activities and Assignments:

- **Weekly reading/viewing** Weekly readings provide the background knowledge, terminology, and examples you need to understand and apply fundamental course concepts. You must complete/view all assigned readings, presentations, and demonstrations included in the lessons. All materials should be completed by the due dates specified.

- **Self-checks** When available, complete self-checks about the online lecture material designed to enhance your understanding and ability to correctly apply concepts covered in weekly readings and presentations. The grading is based on how many self-check questions you have answered correctly in the *first* self-check you submit for the module. Getting a few questions wrong does not result in any deduction for your final grade, unless it looks like you are guessing. Notice that you must complete the self-check for a module *by midnight on Sunday before the module is discussed*.

- **Exam (tentative information; subject to potential change until week 5 of the semester)** You will complete an exam designed to test your understanding of the course concepts. The exam is **closed-book**, i.e., you cannot bring any material, but you need to bring either (1) a **computer or tablet with a keyboard and mouse** or (2) a pencil or pen and a **smartphone** or similar (i.e., a device that can take photos and upload them to Gradescope) to take the exam. Students must be present in the lecture room for the exam. Exceptions are possible for students with disabilities who can provide an official letter from the corresponding Northeastern office at the beginning of the semester.

- **Homework/project** You will complete multiple homework assignments that give you the opportunity to program code and practice the concepts you learn. More

information about these assignments and the course project is available in Canvas.

Course Grading Criteria:

- Self-checks: 5%
- Participation: 15%
- Exam: 60%
- Homework/project: 20%

Class Schedule / Topical Outline: (This schedule is subject to updates.)

Module	Topics	Assignments
1	Trends, Cloud Computing, Parallel Processing Basics	Begin Homework 1
2	Distributed Services: Distributed File System, Resource and Application Management	**Homework 1 due**
3	MapReduce and Spark Overview	Begin Homework 2
4	Fundamental Techniques	**Homework 2 due**
5	Joins	Begin Homework 3
6	Common Algorithm Building Blocks	**Homework 3 due**
7	Graph Algorithms	Begin Homework 4
8	Data Mining 1 (K-Means, Decision Trees)	**Homework 4 due**
9	Data mining 2 (Ensembles)	Begin Project
10	Intelligent Partitioning	
11	More About Spark	**Project Progress Report due**
12	**Exam**	
13	CAP, HBase, and Hive; Flexible Topics	
14	Flexible Topics	**Project reports due**
15	**Project Presentations**	

How to Succeed in this Course

This is an advanced graduate course about an evolving topic. It is therefore essential that you go through the online material carefully and methodically, attend the lectures and participate in online discussions. Homework is designed to help you understand the material and prepare for the exam. The following often works well:

1. When going through the online material, make notes about questions you have or about material you find difficult to understand. Then share these questions through the online forum or in class.
2. When you get a question in a check-your-knowledge quiz wrong or were not sure about the answer, go back to the corresponding online material and try to find the answer.
3. After going through an online lecture, try to explain the material to yourself or to a friend. This way you can better judge if you understand it. Once you identified things that need clarification, try to find the answer yourself by consulting one or more of the recommended books. If you cannot find the answer with reasonable effort, ask others for help (online discussion forum, office hours, and in-class discussions).
4. Start working on homework assignments as soon as they come out. This way you have time to ask questions and get help.

Is This the Right Course for You?

This really is an *algorithms* course at heart. You will write plenty of code, but the main emphasis is on learning how to approach big-data analysis problems. You will need solid Java or Python programming skills to succeed, but we are not teaching any Java/Python basics in this course. You do not need advanced Scala skills and should be able to pick up what you need on-the-fly with reasonable effort.

- If you believe that programming in Java or Scala presents an insurmountable barrier for you, contact the instructor during the first week of classes to find a solution. It is possible to program in other languages, but we generally cannot promise any support for them—so you may be on your own if you get stuck. Students in the past completed their homework successfully using Python for both MapReduce and Spark. Python is well supported in Spark and the programs often look similar to those written in Scala.

We are learning about new techniques that are only partially understood and explored by the research community. Hence in many cases there are no “certain truths.” At times we might find better solutions that could be publishable in a research paper.

We are working with complex cutting-edge software from the open-source community. This means that there will be bugs, lack of documentation, and simply inexplicable behavior at times. Hadoop and Spark also keep changing and updating their API, therefore some code you find in books or on the Web might be outdated or use deprecated features.

When dealing with big data in a complex environment such as MapReduce/Spark and the cloud, developing and debugging code is different compared to traditional settings. Sometimes a task might appear easy but turns out to be much harder and more time-consuming (or the other way round).

You should only take this course if you are prepared to deal with such issues and are willing to put in extra time when necessary. Do not take this course if you want a well-polished and well-tested course without any uncertainty. If you are genuinely interested in the topic and are ready to work around the inevitable frustrations, then this will be a rewarding experience.

Special Accommodations: If you have specific physical, psychiatric or learning disabilities that may require accommodations for this course, please contact Northeastern's Disabilities Resource Center (DRC) at (617) 373-2675. The DRC can provide you with information and assistance to help manage any challenges that could affect your performance in the course. The University requires that you provide documentation of your disabilities to the DRC so that they may identify what accommodations are required, and arrange with the instructor to provide those on your behalf, as needed.

If the Disability Resource Center has formally approved you for an academic accommodation in this class, please present the instructor with your "Professor Notification Letter" *during the first week of the semester*, so that we can address your specific needs.