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: TBD (a course announcement will be posted in Blackboard with this information)
Email: m.riedewald@northeastern.edu
T.A.: TBD (a course announcement will be posted in Blackboard with this information)

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 have to provide evidence in order to claim such an emergency and you have to inform the instructor *as soon as possible*. The following are examples for situations that do *not* qualify as emergencies:
    - I have an interview scheduled.
    - My other course has an exam.
    - 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 (see http://www.northeastern.edu/osccr/academicintegrity/index.html) 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.
    - Violations will be reported to OSCCR.

Course Prerequisites: CS5800, CS7800, or instructor consent. In general, this course is not recommended for first-year masters students who have not taken either of these courses.

Course Description: This course is about techniques for processing big data using many processors. Analyzing big data in a cost-efficient manner has driven the development of novel programming models and system architectures. Not surprisingly, some of the world's leading tech companies including Google, Yahoo, Amazon, Facebook, and Microsoft are at the forefront of this development.

Course Format & Methodology: This course runs for a total of 15 weeks and is delivered

online via the Northeastern Blackboard system accessible at: northeastern.blackboard.com. Each week (or module) contains one or more lessons that you begin on Monday and complete by Sunday of the same week, which is the week *before* the module is discussed. **Please note that all due dates and times are specified according to the local Boston time (Eastern US time zone);** plan to complete and submit all assignments accordingly.

Recommended Textbook & Materials: To gain a deeper understanding of the material covered in this course, we recommend the following books, most of which are available online (and for free) for Northeastern University students from Safari Books Online at http://proquest.safaribooksonline.com.ezproxy.neu.edu/:

- *MapReduce Design Patterns* by Donald Miner and Adam Shook
- *Hadoop: The Definitive Guide* by Tom White
- *Spark in Action* by Petar Zecevic and Marko Bonaci
- *Programming Elastic MapReduce* by Kevin Schmidt and Christopher Phillips
- *HBase: The Definitive Guide* by Lars George
- *Programming Hive* by Edward Capriolo, Dean Wampler, and Jason Rutherglen
- *Hadoop in Action* by Chuck Lam

For some topics we will work with research papers or other online resources. Other important resources will be 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 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 a problem, assigning data to many machines, and then performing local computation in parallel on these 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.
    - We will cover MapReduce (Java) and Spark (Scala).
    - We will discuss HBase, a scalable NoSQL database option for storing and managing big data as key-value records.
- Get hands-on practice writing actual code and running it on many processors.
    - We will work with the Hadoop MapReduce and Spark.
    - We will use the Amazon Cloud to run the code. **You need to have your own Amazon Web Services (AWS) account** to do this, for which you might need to register with your personal credit card. Amazon offers up to $100 free

credit, so be sure to explore this option. However, there is no guarantee that Amazon will give out this credit.

- 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 in CCIS and ECE on related topics. Also note that new approaches for big data processing keep appearing, many 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/or clarification on course material from the instructor are provided throughout the term. Those students who struggle with the material, but take advantage of self-checks and opportunities provided for instructor help and/or peer-to-peer mentoring, can be successful in this course.

The following activities count towards class participation:

1. Asking or answering questions in class, during our classroom time (hybrid version only).
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: Guidelines for completing and submitting each assignment are posted along with the assignment in Blackboard. Late and early homework submission policies will be announced with the individual assignments.

Course Activities and Assignments: This course includes the following required activities and assignments:

- **Weekly reading/viewing** Weekly readings and multimedia presentations provide the background knowledge, terminology, and practical examples you need in order to understand and correctly apply fundamental course concepts. You are responsible for completing the assigned readings and for viewing the presentations and demonstrations included in the lessons. All materials should be completed in the order in which they are presented, and by the due dates specified, within the weekly module.

- **Self-checks** Each week, you complete required self-checks embedded in the online lecture material designed to enhance your current 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, unless it looks like you are guessing. Notice that you have to complete the self-check for a module *by midnight on Sunday, **before** the module is discussed*. As a rule of thumb, if you have carefully studied the material and made a serious attempt to answer all the questions, then you will earn full marks. Complete each self-check as often as you like to ensure you are correctly understanding and applying the course content.

- **Exam** 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 other than a writing instrument. Students in hybrid sections of the course have to be present in the lecture room for the exam. Online students on other campuses have to attend the proctored exam there in person at the announced date and time.

- **Homework/project** You will complete multiple homework assignments that give you the opportunity to apply the concepts you learn. More information about these assignments and the course project is available in Blackboard.

Course Grading Criteria:

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

Class Schedule / Topical Outline:

Please note: for more information about specific assignments and due dates, see instructions within your course site. This schedule is subject to updates; check Blackboard for announcements that will detail any changes.

Module	Dates	Topics	Assignments
1	1/8 – 1/14	Trends & Cloud Computing	
2	1/15 – 1/21	Parallel Processing Basics	Begin Homework 1
3	1/22 – 1/28	MapReduce and Spark Overview	**Homework 1 due**
4	1/29 – 2/4	Fundamental Techniques	Begin Homework 2

5	2/5 – 2/11	Basic Algorithms	**Homework 2 due**
6	2/12 – 2/18	Graph Algorithms	Begin Homework 3
7	2/19 – 2/25	Basic Algorithms, Advanced Applications	**Homework 3 due**
8	2/26 – 3/4	Spark	Begin Homework 4
9	3/12 – 3/18	Intelligent Partitioning	**Homework 4 due**
10	3/19 – 3/25	Data Mining 1	Begin Homework 5
11	3/26 – 4/1	Data Mining 2	**Homework 5 due**
12	4/2 – 4/8	**Exam (on 4/5)**	Begin Project
13	4/9 – 4/15	Databases	
14	4/16 – 4/22	HBase & Hive	**Project reports due**
15	4/23 – 4/29	**Project Presentations**	

How to Succeed in this Course

This is an advanced graduate course about a rapidly evolving topic. It is therefore essential that you go through the online material carefully and methodically, attend the lectures (hybrid version) 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 (hybrid version).
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 (Java, Scala) code, but the main emphasis is on learning how to approach big-data analysis problems. You will need solid Java programming skills to succeed, but we are not teaching any Java 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 relatively little effort.

We are learning about novel 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 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 APS, 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 AWS, developing and debugging code is quite 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 as early as possible.

Northeastern University Copyright Statement

any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the express prior written permission of the University.