Course Syllabus for SIADS 515: Efficient Data Processing

Course Overview and Prerequisites

This course will introduce students to the basics of the linux command-line interface, debugging concepts, basic algorithmic principles such as memoization, recursion, caching, and generators, as well as efficiency and code profiling. There are no prerequisites.

Instructor and Course Assistants

Instructor: Chris Teplovs, Ph.D. Lecturer IV in Information and Research Investigator, School of Information Course Lecturers: Kris Steinhoff, Intermittent Lecturer in Information, School of Information and Staff Software Engineer, Woven Planet

Naga Sanka: MADS Graduate, Intermittent Lecturer, School of Information and Senior Engineer at Stellantis

Course Communication Expectations

If you have questions about course content (e.g. lecture videos, quizzes, or assignments), please use the class Slack channel to discuss with classmates and the instructional team. Instructor and course assistant response time to Slack messages will be within 24 hours.

Required Weekly Textbooks and Readings

See the Library's access instructions for the readings hosted by O'Reilly Online.

Week 1: Linux Command-Line Interface

The Linux Command Line by William Shotts (Chapters 1-4 only)

Week 2: Iterators, Generators, Decorators

Iterators and Generators

Effective Python: 90 Specific Ways to Write Better Python by Brett Slatkin (Chapter 4. Comprehensions and Generators)

Decorators

Head First Python, 2nd Edition Second Edition by Paul Barry (Chapter 10. Function Decorators: Wrapping Functions)

Week 3: Debugging

The Visual Python Debugger for Jupyter Notebooks You've Always Wanted by David Taieb

Python Scope & the LEGB Rule: Resolving Names in Your Code by Leodanis Pozo Ramos

Week 4: Complexity and Optimization

A Common-Sense Guide to Data Structures and Algorithms by Jay Wengrow (Chapter 3)

Python Data Science Handbook by Jake VanderPlasw (Chapter 1, section 7: Profiling and Timing Code)

Learning Outcomes

Familiarity with the linux command-line interface

Understanding generators, memoizing and cacheing

Debugging

Efficient data structures

Theory and measurement of complexity

Course Schedule

This course begins on **Wednesday, March 1st** and ends on **Tuesday, March 28, 2023**.

Weekly **assignments will be due on Mondays at 11:59 pm** (Ann Arbor, Michigan time-Eastern Daylight Time - EDT, UTC -4).

In **Week 1**, you will be introduced to the linux command line interface (CLI) and you will learn basic data manipulation techniques without having to use Python.

In **Week 2**, you will learn about iterators, generators, decorators, caching, and memoization -- all useful techniques to improve the writing and execution of your code.

In **Week 3**, we will review some basic data structures, learn to use advanced ones to help with data manipulation, and debugging techniques for data science.

In **Week 4**, we will discuss algorithmic complexity and look at how we can use profiling to improve the efficiency of our code.

Weekly Office Hours via Zoom (Ann Arbor, Michigan time):

Your instructor will hold weekly, synchronous office hours using the video-conferencing tool, Zoom. The schedule of office hours can be found by clicking on the **Live Events** link in the lefthand navigation menu. Additionally, all office hours will be recorded and archived so that you can retrieve at a later date. Archived office hours can be found in respective module of the course. **Passcode is 515** for all office hours.

Grading

Course Item	Percentage of Final Grade	Due
Week 1 Assignment	25%	Tuesday, March 7th, 2023
Week 2 Assignment	25%	Tuesday, March 14th, 2023
Week 3 Assignment	25%	Tuesday, March 21st, 2023

Week 4 Assignment	25%	Tuesday, March 28th, 2023
Total	**100%**	

Note: All assignments are required to earn credit for this course.

Letter Grades, Course Grades, and Late Submission Policy

We realize that the occasional crisis might mess up your schedule enough to require a bit of extra time in completing a course assignment. Thus, we have instituted the following late policy that gives you a limited number of flexible “late day” credits.

You have three (3) free late days to use during SIADS 515. One late day equals exactly one 24-hour period after the due date of the assignment (including weekends). No fractional late days: they are all or nothing. Once you have used up your late days, 25% penalty for each subsequent 24-hour period after the deadline that an assignment is late. For example, if the due date is 11:59pm Sunday, with no late days left, penalties would be:

Before 11:59pm Wednesday: 25% deduction

Before 11:59pm Thursday: 50% deduction

Before 11:59pm Friday: 75% deduction

After 11:59pm Saturday: 100% deduction

You don't need to explain or get permission to use late days, and we will track them for you. Note that resubmissions after the deadline will be counted as late submissions.

The grading scale for this course is as follows:

A	95%

A-	90%
B+	87%
B	83%
B-	80%
C+	77%
C	73%
C-	70%
D+	67%
D	63%
D-	60%
F	0%

Academic Integrity/Code of Conduct

Refer to the Academic and Professional Integrity section of the UMSI Student Handbook. (access to Student Orientation course required).

Accommodations

Refer to the Accommodations for Students with Disabilities section of the UMSI Student Handbook.

Use the Student Application Form in Accommodate to begin the process of working with the University's Office of Services for Students with Disabilities.

Accessibility

Refer to the Screen reader configuration for Jupyter Notebook Content document to learn accessibility tips for Jupyter Notebooks.

Help Desk(s): How to get Help

- Degree program questions or general help - umsimadshelp@umich.edu
- Coursera's Technical Support (24/7) - https://learner.coursera.help/

Library Access

Refer to the U-M Library's information sheet on accessing library resources from off-campus. For more information regarding library support services, please refer to the U-M Library Resources section of the UMSI Student Handbook (access to the Student Orientation course required).

Student Mental Health

Refer to the University's Resources for Stress and Mental Health website for a listing of resources for students.

Student Services

Refer to the Introduction to UMSI Student Life section of the UMSI Student Handbook (access to the Student Orientation course required).

Technology Tips

- Recommended Technology
- This program requires Jupyter Notebook for completion of problem sets and Adobe or other PDF viewer for reading articles.
- Working Offline
- While the Coursera platform has an integrated Jupyter Notebook system, you can work offline on your own computer by installing Python 3.5+ and the Jupyter software packages. For more details, consult the Jupyter Notebook FAQ .