

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, Lecturer IV in Information and Research Investigator, School of Information • VProv Academic Innovation - Faculty and Staff

Course Lecturers: Joshua Haskins, Intermittent Lecturer, School of Information

Naga Sanka, Intermittent Lecturer, School of Information

Kris Steinhoff, Intermittent Lecturer in Information, School of Information and Staff Software Engineer, Woven Planet

Josh Haskins, Intermittent Lecturer, School of Information

Elham Amini, Intermittent Lecturer, School of Information

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 Opens in a new tab](#) for the readings hosted by O'Reilly Online.

Week 1: Linux Command-Line Interface

[The Linux Command Line Opens in a new tab](#) by William Shotts (Chapters 1-4 only)

Week 2: Iterators, Generators, Decorators

Iterators and Generators

[Effective Python: 90 Specific Ways to Write Better Python Opens in a new tab](#) by Brett Slatkin (Chapter 4. Comprehensions and Generators)

Decorators

[Head First Python, 2nd Edition Opens in a new tab](#) 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 Opens in a new tab](#) by David Taieb

[Python Scope & the LEGB Rule: Resolving Names in Your Code Opens in a new tab](#) by Leodanis Pozo Ramos

Week 4: Complexity and Optimization

[A Common-Sense Guide to Data Structures and Algorithms Opens in a new tab](#) by Jay Wengrow (Chapter 3)

[Python Data Science Handbook Opens in a new tab](#) 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 **Tuesday, March 5th**, and ends on **Monday, April 1st** .

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%	Monday, March 11th, 2023
Week 2 Assignment	25%	Monday, March 18th, 2023

Week 3 Assignment	25%	Monday, March 25th, 2023
Week 4 Assignment	25%	Monday, April 1st, 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 Monday, with no late days left, penalties would be:

Before 11:59pm Thursday: 25% deduction

Before 11:59pm Friday: 50% deduction

Before 11:59pm Saturday: 75% deduction

After 11:59pm Sunday: 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 Opens in a new tab](#) section of the UMSI Student Handbook. (access to Student Orientation course required).

Accommodations

Refer to the [Accommodations for Students with Disabilities Opens in a new tab](#) section of the UMSI Student Handbook.

Use the Student Application Form [in Accommodate Opens in a new tab](#) 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 Opens in a new tab](#) 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/ Opens in a new tab](https://learner.coursera.help/)

Library Access

Refer to the [U-M Library's information sheet Opens in a new tab](#) on accessing library resources from off-campus. For more information regarding library support services, please refer to the [U-M Library Resources Opens in a new tab](#) 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 Opens in a new tab](#) for a listing of resources for students.

Student Services

Refer to the [Introduction to UMSI Student Life Opens in a new tab](#) 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 Opens in a new tab](#).