SC2XX: Scientific Computing

August 30, 2024

Table of contents

About

This is a homepage for *SCI2XX: Scientific Computing* at Southern New Hampshire University, while it is in development. The development of this course is in service of multiple goals:

- to provide SNHU students with exposure to computational tools for solving problems numerically
- to provide SNHU students with an early exposure to the importance of transparent and reproducible analyses
- to provide SNHU students with a foundation in collaborative version control, via GitHub
- to be a foundational course in the three-course certificate in *scientific computing* for students across a wide array of disciplines and who want to approach their field from a quantitative lens.

This project is being led by Dr. David Guo (Aeronautical Engineering) and Dr. Adam Gilbert (Mathematics).

Course Objectives/Competencies

At current, we have three main course objectives.

1. **Develop Skills in Transparent and Reproducible Analyses:** By the end of the course, students will be able to use Quarto Notebooks and Python to conduct and document transparent, reproducible analyses, effectively communicating their methods and results through integrated narratives and multiple output formats.
2. **Apply Computational Methods to Solve Real-World Problems:** Students will demonstrate the ability to apply computational techniques and tools to analyze and solve practical problems, enhancing their understanding of key concepts in mathematics, the sciences, and engineering through hands-on projects and assignments.
3. **Utilize Version Control and Collaboration Tools:** Students will gain proficiency in using GitHub for version control, project management, and collaboration, enabling them to work efficiently on group projects and contribute to open-source scientific computing projects.

List of Topics

Additionally, we have the following list of proposed topics at this time.

Class Number	Topic	Assignment
1	Introduction Software Overview Create GitHub Account	Software Setup: ~ Python ~ Git ~ Positron IDE
2	Link Positron and GitHub My first repo My first Quarto document ~ YAML basics (title and output format) ~ Text "cells" ~ Code "chunks" ~ Rendering	Quarto YAML and Markdown Practice
3	Python Basics Crash-Course (I) ~ Arithmetic ~ Variable assignment ~ Calculations with variables ~ Data types ~ Working with lists	Comment Quarto Notebook from Class and Commit/Push to GitHub

Class Number	Topic	Assignment
4	Python Basics Crash-Course (II) ~ Importing Python Modules ~ `{numpy}` – arrays and operations ~ Vectorized Operations ~ `{pandas}` – data frames ~ `{matplotlib}` and `{seaborn}`	
5	Quarto, YAML, and Markdown Crash-Course (I) ~ Additional output formats ~ Author(s), date(s), themes, and more ~ Headers, paragraphs, and free-text ~ Bulleted and numbered lists ~ Callout boxes	
6	Python Basics Crash-Course (III) ~ Conditional statements ~ Filtering and subsetting ~ Looping (`for` and `while`)	**HW 1: Quarto and Python Foundations**
7	Quarto, YAML, and Markdown Crash-Course (II) ~ Global and local code controls ~ Callout boxes ~ Inline code	
8	GitHub Pages Portfolios (I) ~ Create a `username.github.io` repo ~ Activated *pages* on *main* ~ Edit `index.md` to outline landing page	Add to your GitHub pages site
9	GitHub Pages Portfolios (II) ~ Hyperlinks ~ Linking to rendered reports ~ Creating and including more pages	
10	Creating your own Python functions	
11	Data Visualization (I)	
12	Data Visualization (II)	
13	Numerical Solutions to Algebraic Equations	

Class Number	Topic	Assignment
14	Numerical Solutions to Systems of Equations	
15	Case Study: Applications of Numerical Solutions to Equations and Systems	
16	Case Study (Cont'd): Applications of Numerical Solutions	
17	Curve-Fitting	
18	Case Study: Applications of Curve-Fitting	
19	Case Study (Cont'd): Applications of Curve-Fitting	
20	Optimization and Linear Programming Problems	
21	Optimization and Linear Programming in Python	
22	Case Study: Applications of Optimization and Linear Programming	
23	Case Study (Cont'd): Applications of Optimization and Linear Programming	
24	Simulations and Agent-Based Models (I)	
25	Simulations and Agent-Based Models (II)	
26	Case Study: Applications of Simulations	
27	Case Study (Cont'd): Applications of Simulations	
28+	Final Projects and Debrief	

Additional Resources and Information

We'll use this space to share additional resources and information associated with the Scientific Computing *course* and *certificate*. Each of the items listed below is a living draft and will continue to be updated.

- A draft *Informational Sheet* to help potential partner disciplines decide if they'd like

to pursue a track through the eventual Scientific Computing Certificate for students in their degree programs.
- A short presentation on the state of our progress as of September 18, 2024.

Next Steps

We have the following goals remaining from our Summer 2024 list

1. Finalize a draft syllabus for SC200 (*Introduction to Scientific Computing*)
2. Finalize our current draft "Informational Sheet" so that we can begin sharing it with potential partner disciplines

In addition to those goals listed above, we have the following new goals to be pursued throughout the Fall 2024 semester.

1. Begin discussions, using the "Informational Sheet" as a starting point, with faculty from other disciplines about whether their students could benefit from this course experience
2. Help those faculty identify candidate courses within their degree program that could serve as a suitable *second-tier* course in our ultimate certificate in *Scientific Computing*
3. Prepare for the SC200 course to go through the academic governance pipeline
4. Investigate the feasibility of developing a three-course *certificate* and what structural barriers we may face in that effort