Discussion Group: List of Best Practices for the class

Ten Simple Rules for Quick and Dirty Scientific Programming (G Balaban et al., 2021)

- Think before you code – Gather all useful information before programming.
- Start with prototypes and expand them in short development cycles.
- Look for opportunities for code reuse.
- modularize your code – Organize into functions or classes that are evolvable.
- Avoid premature optimization.
- Refractor the code by improving the internal structure of the source code without altering its external behavior.
- Write self-documenting(variable and function names used in the codes are explanatory) code for programmers and a readme file for users.

Best Practices for Scientific Computing(G Wilson et al., 2014)

- Code should be written for people not computers.
- Program should be robust and suited for repeated workflows.
- Make Incremental Changes
- Don't Repeat Yourself (or Others)
- Add checkpoints in your program to see what you're doing (error trapping).
- Optimize software only after it works correctly.
- Make sure to use descriptive comments instead of general statements.
- Collaborate with others – a fresh set of eyes can help catch errors.

Good Enough Practices in Scientific computing (G Wilson et al., 2017)

- Data management: Always save data in its original form.Document the extraction and cleaning process and make it available in a public repository.
- Software: Include brief explanatory comments, eliminate duplication,give functions and variables meaningful names and submit code to a reputable DOI-issuing repository
- Project Organization: organizing the digital artifacts of a project to ease discovery and understanding
- Tracking changes: Back up everything created,Use a version control system.

Principles for data analysis workflows(Stoudt et al., 2021)

Follow a three step software design and development principles -**E R P**- for every data intensive decision science project:

- **E**xplore: Explore data, explore a potential solution and survey the software tools or libraries available. Develop a strategy to clean and analyze data.

- **R**efine: Finalize suitable analytical approach, design of experiment, develop prototype and build reproducible codes for results.

- **P**roduce: This phase happens concurrently with Explore and Refine phases but focus here is to standardize the code and documentation and prepare the work for wider consumption.

Summarized by: Chris Hoang, Francis Kurian, Nojoud Almohanna, Simran Gaglani