Department of Mathematics, University of Utah
Topics in Numerical Solutions of PDE
MTH6630 – Section 001 – Spring 2023

Project submission guidelines
Basics of submission

Submission instructions

You are required to submit all projects via the version control system Git. You will do so by "pushing" updates to a remote repository. For each assignment, you will need to,

1 Write a report detailing your solution to the excercise(s). Your report should briefly discuss the assignment, your solution, and document results based on the requested simulations above. When I disseminate the assignments, I will provide a LaTeX template for submission.

2 Create a git repository on your computer (the "local" repository) that contains <u>both</u> a LaTeX report with figures, <u>and</u> the source code that reproduces the figures as they appear in the report. (Any programming language is fine.)

3 Create an account on Github (`github.com`).

4 Create a (private) repository named `math6630-project-X` on your Github account (the "remote" repository), where `X` is the assignment number. In your local repository, create a pointer to the remote repository with the `git remote` command.

5 Submit your project to the remote repository with a `git push` command from your local repository. (You might need to create an SSH public/private keypair and upload your public key to bitbucket to accomplish this.)

6 Give me (username `akilnarayan`) write access to your `math6630-project-X` Github repository. At this point, I have access to your submission.

7 On the assignment due date, I will confirm receipt of your submission by pushing a tag to your repository.

Submission expectations

There are two components of the submission: a LaTeX report and source code. <u>Both</u> are mandatory.

<u>General repository etiquette</u>

- If there are unusual things about the way you've organized your files, provide a `README` text file that documents the general structure of the repository. E.g., if I cannot look at the files in your repository and <u>immediately</u> understand where the LaTeX report and source code are, and also how to compile them, then you need a README. I suggest that the base folder should contain a subfolder called `code` that contains *all* code that reproduces figures in your LaTeX document.

- Provide a `makefile` that allows me to easily compile your LaTeX report. If your code is in a compiled language, provide another makefile for compiling your code.

- Your repository should in general only track ASCII-type text files (images are an exception). It should <u>not</u> track compiled binaries, pdf output from tex compilation, or datasets generated

by your code. In general your repositories should be *small* in disk space, with the largest files probably being the pictures used in your reports.

LaTeX report expectations

- Write your report so that someone can understand the assignment without having seen the assignment.

- Extreme verbosity is unnecessary (and should be avoided), but enough detail should be provided so that the general implementation (or your solution for the implementation) is reasonably clear to someone educated in the course material.

- Do <u>not</u> include any source code text in your report.

- Your report is largely an explanation of the mathematics required for your solution, along with a compilation of numerical results.

- Your report should describe the solution in a programming language-agnostic manner. I.e., you should not make reference to any special data structures, code modularity, memory allocation, or syntactical solutions that you had to implement in order to complete the assignment.

- Number all figures and tables, e.g., `Figure 2`, `Table 1`, etc.

Code expectations

- All computer code should be located in the `code` subfolder.

- Include a README text file that explains the purpose of the main files in the subfolder. You should write your README file with the intention of giving a clear guide to a recipient about what they are supposed to do once they receive your code.

- If you use a compiled language, include a `makefile` that generates all necessary binaries.

- Each figure/table in your report should be paired with a file in the `code` subfolder that reproduces that figure/table. E.g., if you are coding in Matlab and your report contains `Figure 2` and `Table 1`, files `figure_2.m` and `table_1.m` should exist in the `code` subdirectory, and running those files should reproduce the figures. (If using a compiled language, binaries named, e.g., `figure_2.o` and `table_1.o` should be generated by your makefile.)

Example submission

You should consider assignment 0 (posted online now) as an example submission. Please check this for a demonstration of the above procedures.