

5 Programming Assignments

- the last PA is optional
- Gradescope: entry code D3G5NK
("group functionality" for teams of 2)
- MATLAB, Python, C, C++
(If another language → contact the TA first)

Grading:

- Program (40 points)
- Results (15 points)
- Report (45 points)

Signed cover sheet

Name 1	
Email	
Other contact information (optional)	
Name 2	
Email	
Other contact information (optional)	
Signature (required)	I (we) have followed the rules in completing this assignment _____ _____

Program (40 points)

- Design and overall program structure (20)
 - readable, efficient, consistent, works
 - Avoid duplicating code
 - Avoid deep nesting
 - Use consistent formatting
 - Use required PROGRAMS, OUTPUT directories --- ZIP / TAR
 - Make sure the code runs!
- Reusability and modularity (10)
 - Use short and focused functions
- Clarity of documentation and programming (10)
 - Use descriptive names for variables and functions
 - Provide enough clear and concise comments (variables' /objects' description, workflow steps)
 - Indicate purpose /objective, inputs, and outputs at the top of each source file
 - Provide "README.TXT" (the names of all source files + short description of each file)
 - Indicate the executable program
 - Provide "Instructions" for running the executable program
 - Indicate authorship of the code sections

PROGRAMS directory:

- All source files (including the executable)
- "README.TXT"
- "Instructions" to run the executable

Results (15 points)

→ Correctness and completeness, OUTPUT directory

- Provide output for debug datasets
- Provide output for unknown datasets
- Make sure the code generates files for the output
- Make sure that the output files' format is correct

Report (40 points) → A narrative!

- Summary of the problems to be solved
 - Description of mathematical approach
 - Description of algorithmic approach
 - Indicate the programming language used
 - Describe the algorithmic steps taken (variables / Pseudocode ≠ copy of the code / Inputs, Outputs)
 - Provide citations for any used libraries / packages
- } Description of formulation and algorithmic approach (15)
- Overview of program structure (10)
 - Hierarchy chart, table, or diagram + discussion
 - Discussion of validation approach / program verification (10)
 - Component-level testing / unit testing using custom-made datasets
 - Entire program testing using given debug datasets
 - Discussion of results (10)
 - Analysis (tabular summary, plots) + discussion of the results for debug datasets
 - Average error between the given and your outputs for each dataset
 - Error analysis
 - A tabular summary of the results for unknown data
 - Statement “Who did what”
 - Citations for Dr. Taylor’s slides and all other used sources! (Points off otherwise. Applicable to the code and report)