General notes

- Late penalty (to be fair towards all students):
  - prorate based on hours (1 day = 24 hours = -10 points)
  - round to one decimal place
- Ethics page (to promote ethical work and healthy teamwork):
  - 0 = present and signed
  - -5 = missing or not signed

PA grading rubric

Program – Design and overall program structure (20):

- 0 = Correct
- -2 = Minor issues in the program structure (e.g., inconsistent indentation, long lines)
- -4 = Some issues in the program structure (e.g., unnecessary loops, deep nesting)
- -10 = Poor program structure (e.g., a lack of logical organization)
- -0.5 = Small source code modifications necessary to run the program
- -1 = Some source code modifications necessary to run the program
- -3 = Source code modifications necessary to run the program
- -3 = The program does not produce output files
- *0 = The program produces files that are different from the ones submitted (notify TA)*
- -6 = The program does not run
- -2 = Do not follow instructions regarding PROGRAMS & OUTPUT directories

Program – Reusability and modularity (10):

- 0 = Correct (short & focused functions, well-defined modules)
- -2 = Some functions are lengthy
- -5 = All functions are lengthy
- -9 = No functions
- -3 = A lack of well-defined modules
- -5 = No well-defined modules

Program – Clarity of documentation and programming (10):

- 0 = Correct

- -1 = Lacking comments in a few parts of the code
- -3 = Some comments, but inadequate to provide clarity
- -5 = No comments in the code
- -1 = Some variables/functions have poor names
- -3 = Variables/functions have poor names
- -1 = Incomplete README file
- -3 = Missing README file
- -1 = Incomplete instructions for running the code
- -3 = Missing instructions for running the code

Results (15):

- 0 = Correct
- -8 = Missing output for debug datasets
- -7 = Missing output for unknown datasets
- -2 = The output is close but incorrect
- -5 = The output is incorrect or incomplete
- -0.5 = The format of output files has a minor issue (e.g., naming, header, or spacing)
- -1 = The format of output files has several issues
- -2 = The format of output files is incorrect

Report – Summary of the problems to be solved (1):

- 0 = Correct
- -0.5 = Incomplete or incorrect summary of the problems to be solved
- -1 = Missing summary of the problems to be solved

Report – Description of mathematical approach (7):

- 0 = Correct
- *Deduct as needed based on the completeness (see details below)*
- -1 = Minor mistakes in the description of mathematical approach
- -0.5 = Missing some citations / references
- -2 = No citations / references provided
- -7 = No description of mathematical approach is provided
- -2 = Implementation of Arun's method does not follow the complete algorithm described in Arun's paper. Ignores corner cases
- -1 = Incomplete Arun's method implementation. Checks for reflection but fails to check for zero singular values

Report – Description of algorithmic approach (7):

- 0 = Correct
- *Deduct as needed based on the completeness (see details below)*
- -0.5 = A pseudocode has programming syntax
- -2 = A pseudocode is the copy of the code
- -1 = Minor mistakes in the description of algorithmic approach
- -0.5 = Missing some citations / references (e.g., for used libraries and packages)
- -2 = No citations / references provided (e.g., for used libraries and packages)
- -7 = No description of algorithmic approach is provided

Report – Overview of program structure (10):

- 0 = Correct
- -2 = Minor mistakes
- -5 = Poor overview of the program structure
- -8 = Significant mistakes
- -10 = Missing overview of the program structure

Report – Discussion of validation approach (10):

- 0 = Correct
- -1 = Minor mistakes
- -2 = Incomplete explanation of testing steps/results
- -5 = Lack of detail on testing steps/results
- -5 = No unit testing
- -10 = Missing discussion of program verification

Report – Discussion of results (10):

- 0 = Correct
- -2 = Incomplete discussion of the results obtained for debug datasets
- -5 = Missing discussion of the results obtained for debug datasets
- -3 = Missing tabular summary of the results for unknown datasets
- -2 = No statement of who did what for a team submission

Late penalty (prorate based on hours):

- 0 = Correct
- -10 = 1 day late
- -20 = 2 days late

Ethics page:

- 0 = Present and signed
- -5 = Missing or not signed

Notes on Programming Assignments for students

General:

- Submit 2 files: Report and Zip file
- If you experience difficulties with uploading a zip file to Gradescope, please email it to the TA **before** the deadline while still uploading the Report to Gradescope
- If you work in a Team, add your partner's name to the submission on Gradescope

Zip file should contain:

- Folder "PROGRAMS" with
  - All source files
  - A folder with input data files
  - "README.TXT" listing the names of all source files with a 1-line description of each file
  - "Instructions.txt" for running your program: note the programming language and its version, list all required packages/toolboxes, note the main file or the executable and explain how to run it

  P.S.: If you used GitHub, please make sure to delete "__MACOSX" folder from your zipped file before uploading it to Gradescope as it tends to cause issues

- Folder "OUTPUT" with
  - Output for debug datasets
  - Output for unknown datasets

  Make sure that the format of your output files is correct.

About Program:

- It should work without the need for any code modifications form the user
- It should generate all output files in one run
- It should be well-structured, organized, and readable (have consistent indentation, relatively short lines, no duplications, no unnecessary loops, etc.)
- It should have short & focused functions and well-defined modules

- It should have good descriptive names for all variables and functions
- It should have enough clear and concise comments in English (the description of variables/objects, workflow steps)
- All modules, functions, classes, methods should have docstrings describing the parameters and return type
- It is a good practice to indicate authorship of the code sections

About Report:

- There is no strict length limit but usually 10-15 pages is enough
- Please submit PDF of the report with no watermark on the pages
- Fill-out the assignment page, sign it and make it the first page of your report
- The report must be a narrative
- The report should have the following CLEARLY DEFINED and DISTINCT sections (please, do not combine the sections as it makes it more difficult to grade):

I. Introduction

Provide a summary of the problems to be solved. Do not copy and paste the sentences from the assignment. Instead, describe the assignment objectives in your own words. If you use a picture from the assignment made by the professor, please cite it (read about the citations below)

II. Description of mathematical approach

- Describe the objective and mathematics behind the assignment steps
- Make sure to cite Dr. Taylor's slides and other used sources
- Add in-text citations and the full reference list at the end of your report
- Make sure to provide all necessary formulas in this section. It is better to type the formulas yourself instead of taking screenshots of the professor's slides while still providing a citation. The description of all variables used in the formulas is important
- Figures are recommended but not required
- You should re-use relevant descriptions from your previous PA reports (if applicable) rather than refer the graders to your previous submissions because we do not have time to switch between the assignments for every student. Furthermore, some students change partners, programming language, methods.

III. Description of algorithmic approach

- Indicate the programming language used

- Describe the algorithmic steps taken for all the steps mentioned in the math section. Here, we would like to see the functions you created, their purpose, input and outputs. You should also describe the workflow of the functions either in words or via a pseudocode (which is **not** the copy of the code) where you can reference relevant formulas (numbered) described in the math section to keep the report concise
- Provide citations for all used libraries / packages
- You should re-use relevant descriptions from your previous PA reports (if applicable) rather than refer the graders to your previous submissions because we do not have time to switch between the assignments for every student. Furthermore, some students change partners, programming language, methods

IV. Overview of program structure

Here, would like to see a hierarchy chart showing the relationship between different functions/modules/units of your program. A table might also work, although it is not as illustrative and informative as a hierarchy chart. In any case, besides the visual representation of your program structure, add a short discussion of it emphasizing the reusability and modularity of your program.

If you have described all functions (names, inputs, outputs, purpose) and units in the algorithmic steps, add a sentence indicating that, so that the reader would know where to look for the details. Otherwise, make sure to complete the description of the remaining program components here.

V. Discussion of validation approach

- Discuss unit testing using your own custom-made datasets:
  - Describe the rationale and procedure (in words or pseudocode) of the custom data generation and note the functions created (names, purpose, inputs, outputs)
  - Describe the testing itself and the outcomes you observed. Add some visualizations (plots, tabular summaries, screenshots) to make your discussion more informative and solid
- Show the results (tabular summary, plots) of the entire program testing using the provided debug data. It is up to you how to compare your output with the given one. Discuss your approach so that the reader can understand

VI. Discussion of results

- Discuss the difference between the given and your outputs for debug datasets
- Provide a tabular summary of the results for unknown data

VII. Statement "Who did what"

Discuss who did wat for the code and the report, in sufficient detail.

What to discuss in mathematical and algorithmic sections for different PAs

PA1:

- Your selected 3D point cloud to 3D point cloud registration
- Pivot calibration
- Calculation of the expected EM marker positions on the calibration object
- EM probe pivot calibration
- Optical probe pivot calibration

PA2:

- Your selected 3D point cloud to 3D point cloud registration
- Calculation of C expected values
- Your selected Distortion correction function
- EM probe pivot calibration using the distortion correction function
- Calculation of the locations of fiducial points with respect to the EM tracker base coordinate system
- Calculation of the registration frame
- Calculation of the pointer tip location with respect to CT image