

ISTA 410/510

General Homework Instructions

For contribution to the final grade and due dates see the course and/or assignment web page.

1. Handing in your work

Every completed assignment will consist of a PDF and code for all programming parts of the assignment. The TA will spot check the code. However, the PDF must stand alone as a record of your work, and it should be written as though the instructor does not have access to the code.

Instructions for handing in work are under construction.

2. Programming language

Except where indicated otherwise (e.g., the first assignment must be done in Matlab), programming work can be done in any language you like. However, you are encouraged to learn Matlab as part of this course. Note that if you choose to use a language that the TA does not know, then they will not be able to help you, and they will be less well equipped to look at your code and give you part marks. You can assume that the TA knows Matlab and C/C++.

3. Assignment question weights, hints, bonus marks, etc.

For simplicity, problems are generally all worth the same, except ones marked by “+” that are expected to substantively more time consuming, and are worth double. Two “+” means triple value, etc.

Questions marked by “@” will not be graded and need not be handed in. However, you are responsible for being able to do them. You should read these questions, think through what you would do for them, and only then consider skipping them if they do not seem to be useful.

Questions marked by “*” are required for grad students, but optional for undergraduates. Problems with two or more “*” are optional. These may be especially challenging, or project-like, or may simply require background that many students do not have.

Any non-challenge problem can be replaced by challenge problems with collective value is at least that of the problem being replace (e.g., undergraduates can exchange a non starred “+” problem with two “*” problems without “+”). Please make it clear that this is what you are doing (e.g., for a required problem you could answer “see optional problem #3”). The point here is to enable students to avoid problems that they feel are not instructive.

Extra problems (please indicated in your answer when you are doing an extra problem) are eligible for modest extra credit (extra credit does not scale linearly and is up to whim of the instructor). The maximum score for an assignment or take-home exam will be capped at 120%. The maximum score for all assignments taken together is capped at 65/60.

For clarity, deliverables will be flagged with “\$”, but sometimes the “\$” will be missing. If it is clear from the question that something should be handed in, but does not have the “\$”, it is still required. Email the instructor to check, so that he can fix the omission.

Hints for some problems may be provided in advance, or due to popular request. Good form demands seriously thinking about the problem before peeking at the hint. If you make use of a hint, make a note of it. This is a matter of academic honesty.

Similarly, if after a solid effort to do a problem yourself, you get unstuck by consulting with someone or by making use of some other resource, simply make a note of it. For example, you might say that you had a glance at the solution to the same or similar problem solution in a particular source, and then attempted to recreate for yourself. This is better than being completely stuck, or copying the answer blindly, which of course is academic dishonesty.

4. Writing your assignment

As already described, each submitted assignment consists of at least a PDF. The beginning of that PDF should provide any meta information such as any questions you did not do, any questions you substituted, and any questions that you did as extra. If there was programming involved, then you should also describe the languages used (e.g., Matlab), the machine you tested it on (e.g., your Mac laptop), and any non-obvious code directory organization that was not prescribed in the assignment description.

The person grading your assignment will use the PDF as their guide, so any question that includes code that can or should be run by the grader should indicate that (e.g., “See Matlab script file”). Sometimes that will be all that is needed, but most often you will be asked to explain your results in the PDF.

The instructor considers preparing your assignment a form of writing, and writing is hard work! Good writing tells a story efficiently. To do it well, you need to develop an intuition for what the reader wants to know, which is often a bit different from what you want to write.

Where applicable, assignments will be graded roughly 65% for content and 35% for exposition. It is not possible to provide detailed feedback on your exposition as part of the grading process, but the instructor is happy to discuss exposition in person, either with individuals, or as a group.

Obviously, not all answers will require a lot text. Showing your work, or telling the reader what you did (and why) does not always take a lot of space. However, having a good sense of what needs to be said is an important skill worth developing. In some cases, snippets of code may make sense. In other cases, a figure with a good caption will be most of what is needed. And finally, sometimes a very short answer is what is needed.

Checklist (version one):

1. Did you read what you wrote and edit it to make it better?
2. Does your answer actually address the question? Is it obvious that this is the case?
3. Does your answer demonstrate your understanding by showing how you got the result, and why the key steps were taken or are justified? Remember, the instructor generally asks questions for a reason. It may help to think for moment about what that may be.
4. Do your figures have captions? Do the captions tell the reader what they are suppose to conclude from looking at the figure? You do not need to tell the reader what is visually obvious (“the blue bar is taller than the red one”) but you could tell them the main point (“the blue bar is taller than the red one because, on average, people that exercised more reported being happier”).
5. Can the instructor read and understand your answer quickly? If your answer involves writing more than few sentences, are the paragraph breaks sensible? Does the first sentence of the each paragraph indicate the main idea, and whether the rest of the paragraph can be skipped if reading

quickly? If you read the first sentence of each paragraph in sequence, does the flow make sense? (This is a good trick to apply to section and sub-section headings in longer documents).

6. If you read your work objectively as a naive reader, does it make sense? This is sometimes easier to do if you come back to the assignment after a day or two. Of course, this requires starting it early enough. If something is complete and correct (i.e., locally it appears fine), and yet hard to follow, you should reconsider the order that you tell the reader the story.
7. Did you spell check?