



General Requirements for Assignments

Several assignments will be given in this course. So that you know what is expected of your submitted work, I have put together this set of rules and guidelines. They apply to all assignments that I give in this class. If you do not adhere to all of these rules, you may lose points on the assignment. They are designed to facilitate consistent grading.

1. Problems that are stated in the form, “show that...” or “prove that...” and so on, require that you write a **proof**. For these types of problems, you must write a proof. Your answer will be graded on the correctness and clarity of the proof. It is assumed that you know what a proof is, because proofs have been a part of the material in many of the prerequisite courses.
2. If a problem asks for a yes/no answer, or a short numeric answer (such as those of the form *how many...* or *how large...* or any other question whose answer is a number), the answer must include a justification, which may require a proof or some other explanation, depending upon the nature of the question. The point is that I need to see how you derived your solution. This makes it possible to award partial credit for incorrect answers, and lets me see how you thought about the problem. Most of the credit is awarded based on the justification, not the answer.
3. All assignments must be submitted as specified in the assignment’s instructions. Some assignments are submitted through Gradescope whereas others are submitted via the Computer Science Department’s network into the directory specified in the assignment’s instructions. Each assignment will contain detailed instructions for how to do this.
4. Assignments must be submitted as plain text files, unless the instructions state otherwise. No other format will be accepted. In particular, no proprietary formats such as Microsoft Word *doc* or *docx* will be accepted. A plain text file is a file whose type is either “ASCII text” or “UTF-8 Unicode text” and which does not have CRLF line terminators. Microsoft Windows applications that create text files use these terminators; such files must be converted to Unix format using the `dos2unix` command. For example, if a file you have uploaded from a Windows application is named `hwk2.txt`, on any CS network computer, you would type

```
dos2unix hwk2.txt
```

and it would be converted to Unix format.
5. Assignments will not be accepted by email, unless I have specifically allowed it in certain rare cases.
6. All assignments must be submitted on time to receive any credit at all. Late assignments are not accepted.
7. Unless stated otherwise, all questions within an assignment have equal weight.
8. All assignments must comply with the rules regarding Academic Dishonesty stated in the course syllabus.